

12

NRL Memorandum Report 3552

FL

# A Fortran Program to Plot Phase Shifts of the Three-Dimensional Square Well

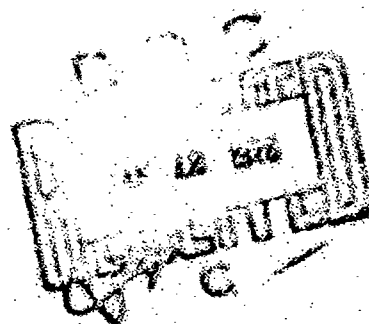
JAMES L. RENACE

Solid State Devices Branch  
Electronics Technology Division

154019116

CONFIDENTIAL TO THE U.S. GOVT  
EXCEPT WHERE SHOWN OTHERWISE

July 1975



NAVY RESEARCH LABORATORY  
Washington, D.C.

Approved for public release; distribution unlimited.



SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NRL Memorandum Report 3082	3. GOVT ACCESSION NO.	2. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A FORTRAN PROGRAM TO PLOT PHASE SHIFTS OF THE THREE DIMENSIONAL SQUARE WELL.	5. REPORT TYPE PERIOD COVERED A final report on one phase of a continuing NRL problem.	
6. AUTHOR James L. Repace	7. PERFORMING ORG. REPORT NUMBER	
8. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Research Laboratory Washington, D.C. 20375	9. CONTRACT OR GRANT NUMBER(s)	
10. CONTROLLING OFFICE NAME AND ADDRESS Office of Naval Research Arlington, VA 22217	11. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NRL Problem R08-42 NRL Thomas Edison Program	
12. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	13. REPORT DATE Jul 75	
	14. NUMBER OF PAGES 46	
	15. SECURITY CLASS. (of this report) UNCLASSIFIED	
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE	
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENTS for the abstract entered in block 20, if different from Report		
18. SUPPLEMENTARY NOTES <b>COPY AVAILABLE TO DDC DOES NOT PERMIT FULLY LEGIBLE PRODUCTION</b>		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) 3 dimensional square well Potential scattering Phase shifts		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This program calculates, and provides CalComp plots of, the CDC 3800 Computer, phase shifts in radians or degrees for quantum-mechanical scattering from a 3-dimensional spherically symmetrical square well potential of width $a$ and depth $-V$ , for a de Broglie quantum particle of energy $E$ . The program, which can plot in two or three dimensional representations, calculates the first five phase shifts, and permits the determination of the number of bound states the well will support for each phase shift.		

DD FORM 1473 1 JAN 73

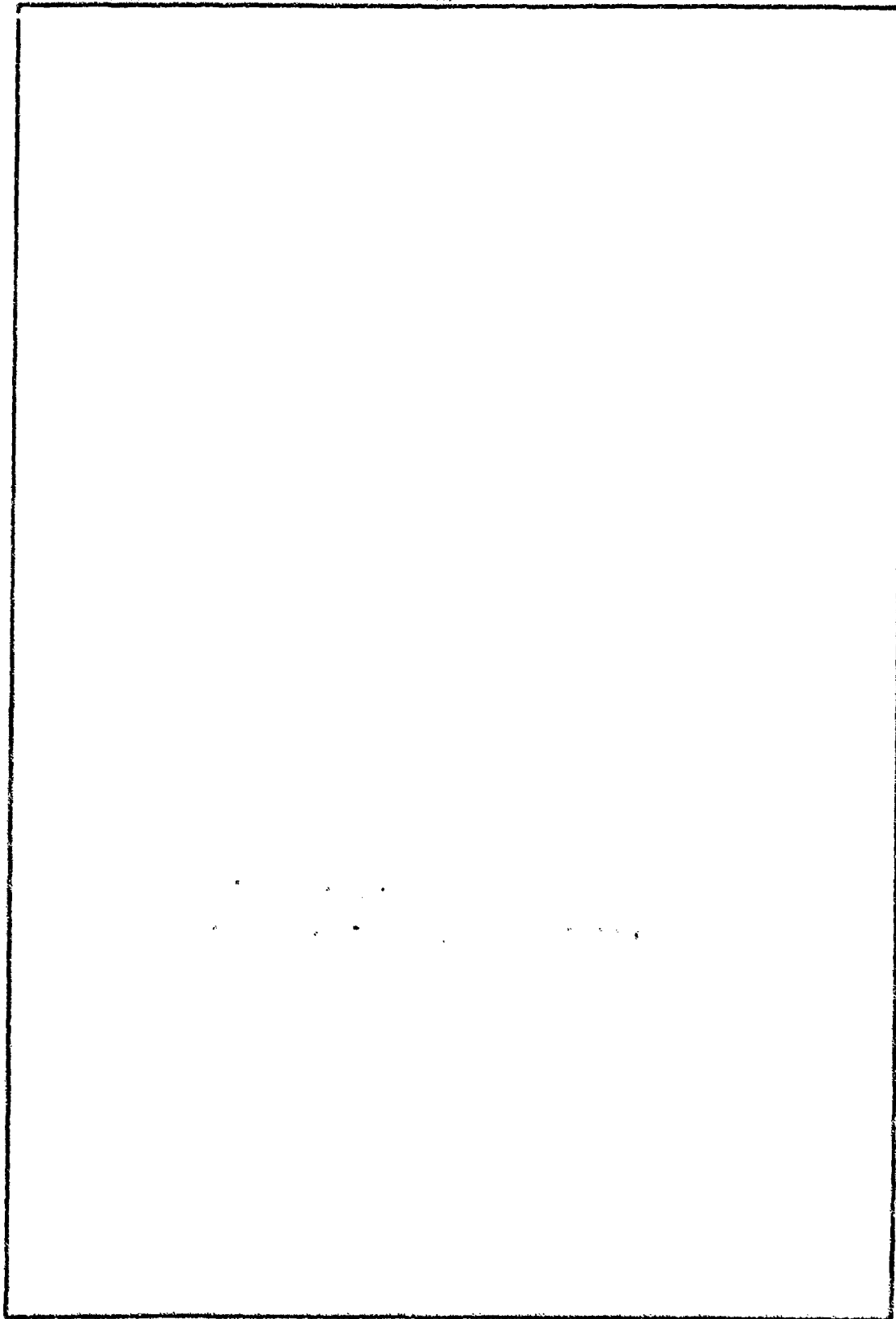
EDITION OF 1 NOV 68 IS OBSOLETE  
S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

251 950

mb

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)



SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

## CONTENTS

INTRODUCTION .....	1
1.0 IDENTIFICATION .....	1
2.0 PURPOSE .....	2
3.0 USAGE .....	3
4.0 METHOD OR ALGORITHM .....	11
5.0 LISTING OF PROGRAM .....	13
6.0 COMPARISONS .....	13
7.0 TEST METHOD AND RESULTS .....	13
8.0 REMARKS .....	14
9.0 ACKNOWLEDGEMENT .....	17

## A FORTRAN PROGRAM TO PLOT PHASE SHIFTS OF THE THREE DIMENSIONAL SQUARE WELL

### INTRODUCTION

This program calculates, and provides CalComp plots of, phase shifts in radians or degrees for quantum mechanical scattering from a 3-dimensional spherically symmetrical square well potential of width  $a$  and depth  $-V$ , for a de Broglie quantum particle of energy  $E$ . It exists in two versions, one of which provides plots of the phase shift  $\delta_l$  vs  $Ea^2$ , with the phase shift given in units of  $\pi$  and  $Ea^2$  in dimensionless units, in a standard two dimensional CalComp plot. The other version adds a third dimension to the plot, with  $\delta_l$  being plotted vs  $Ea^2$  and  $V$ . The program permits the first five phase shifts to be calculated ( $l$  values 0 thru 4) and permits a determination of the number of  $s$ -wave bound states the well will support via Levinson's Theorem.

### 1.0 IDENTIFICATION

#### 1.1 Title

Calculation and plotting routine for determination of phase shifts of the spherically symmetrical square well potential.

#### 1.2 Entry points: Both versions

PHASHIFT (2D and 3D versions)

SPHJW

Note: Manuscript submitted June 25, 1973

SPHYN

MINIMAX

LAYOUT (2D and 3D versions)

1.3 Programming Language

Language: 3600/3600 FORTRAN

Routine Type: Program, Subroutine

Operating System: DRUM SCOPE 2.1

1.4 Computer and Configuration

CDC 3600

1.5 Contributor or Programmer

J. L. Repace, Code 5216J, Solid State Devices Branch, Electronics Technology  
Division

1.6 Contributing Organization

NRL - Naval Research Laboratory, Washington, D.C. 20375

1.7 Program Availability

1.7.1 Submittal: Program write-up, FORTRAN source deck, source listing

1.8 This program has been used to calculate and plot in two and three dimensional perspective the phase shifts of the spherically symmetric square well potential for square wells of depth 0 thru 1000 in dimensionless units, for energies 0 thru 5000 in dimensionless units, for s thru g-wave scattering. (VERIFICATION)

1.9 1 June 1975

2.0 PURPOSE

Although the general properties of the phase shift of the three dimensional spherically-symmetrical square well potential\* are described in most textbooks on quantum mechanics, specific graphical representations of this function are rare.

\*Shown in Fig. 1

This program permits two and three dimensional Calcomp plots of s, p, d, f and g-wave phase shifts, as a function of energy.

Input is handled by PHASHIFT, with the well parameters energy limits and plotting information being read from data cards. The phashifts are then calculated, with PHASHIFT calling SPHYN and SPHJN, routines which respectively compute spherical Neumann and spherical Bessel functions. Then PHASHIFT calls MINIMAX, which computes the largest and smallest phashifts calculated if the plots are to be automatically scaled. Finally, the plotting routine, LAYOUT is called and a Calcomp plot is produced.

### 2.1 Problem Background

In many fields of physics, observations are described in terms of the scattering of an electron, or other particle, on a target represented by a three dimensional square well. Examples are found in nuclear physics, in atomic physics, and in solid state theory. Often, the square well approximation is only used initially to obtain some idea about the size of the scatterer and of the strength of its potential. In many cases the approximation is augmented with a variety of refinements. Nevertheless, the three dimensional square well provides us with the prototype of the behaviour of a scattering object. This is the case in quantum mechanics, and also in electromagnetic theory. Although the general properties of the phase shift of the three dimensional spherically-symmetrical square well potential are described in most textbooks on quantum mechanics, specific graphical representations of this function are rare.

### 3.0 USAGE

#### 3.1 Calling Sequence or Operational Procedure

To use PHASHIFT as a program, data cards after the %LOAD card are required. See section 3.7. There are two different versions, depending upon whether two



dimensional plot displays one through five phase shifts versus the energy-well-width parameter  $Ea$ . The three dimensional plot displays a selected phase shift versus  $Ea^2$  for up to nine different potential depths, where the potential depths are displayed on the third axis, which is oriented at an angle of about  $20^\circ$  from the abscissa. The well depths are displayed equally spaced along the potential axis, although they need not be consecutive.

### 3.2 Arguments, Parameters, and Initial Conditions

#### 2-D Version and 3-D Version

ZERO - Lowest energy value

EPSILON - The increment to ZERO for each iteration, of which there are LAST

LAST - The total number of data points desired ( $\text{EPSILON} \times \text{LAST}$  = the maximum value of the energy)

VULCAN - the well depth

FERMI - the well width

MAX - The highest order phase shift desired (all phase shifts up to MAX are superimposed on one plot)

DEGREE - determines whether the phase shifts are given in degrees or radians

NO PLOT suppresses plotting

NOSCALE suppresses ordinal self-scaling: if self-scaling is suppressed, maximum and minimum values of the phase shift YMAX, YMIN must be inserted prior to statement 100 in PLOTSHIFT

KILL Designated last data card: the position of KILL on the data cards differs between the 2-D and 3-D versions.

For additional information concerning these parameters, see Section 3.7.

### 3.3 Space Required (Decimal and Octal):

#### 3.3.1 Unique Storage:

2-D version:

		Locations
PHASHIFT	12722 octal	
MINIMAX	00230 "	"
SPHJN	01140 "	"
SPHYN	00277 "	"
LAYOUT	01332 "	"
SCALE	00246 "	"
AXIS	00536 "	"

3-D version:

		Locations
PHASHIFT	13003 octal	
MINIMAX	00230 "	"
SPHJN	01140 "	"
SPHYN	00277 "	"
LAYOUT	01332 "	"
SCALE	00246 "	"
AXIS	00536 "	"

3.3.2 Common Blocks:

Both Versions:

COMMON/LABEL/IL, ILX(5), ILY(5), ILH(5), ILR(5), ILC(5), ILABEL(5,5)  
COMMON/RANGE/XMAX,XMIN,YMAX,YMIN,IFRST(5),ILST(5),XSIZE  
(In the 3-D version the subscripts for all variables and arrays in LABEL and RANGE are (9) and (9,9) resp.)  
COMMON/1/DELTA(1000),CASTOR(1000),POLLUX(1000),HERCULES(1000),AJA  
IX(1000),OLYMPUS(1000),HADES(1000),B(1000),DB(1000),N(1000),DN(1000  
2),BZ(1000),DBZ(1000),NZ(1000),GAMMA(1000),ENERGY(1000),CYCLOPS(100  
30),CENTAUR(1000),ZEUS(1000),RHO(1000),DNZ(1000)  
COMMON/A/PHASE(5000),ERGON(5000) (In 3-D version this is called COMMON/3/)

3.3.3 Temporary Storage: None

3.4 Messages and Instructions to Operator: None

3.5 Error returns, Messages and Codes

These are all related to the Bessel and Neumann function subroutines:

See NRL Computer Bulletin 29, section 3.6 for explanation of error messages relating to overflow and underflow prevention and use of zero arg. for Neumann functions.

3.6 Informative Messages to the User

Other than the error messages mentioned in Section 3.5, no informative messages are printed.

### 3.7 Input

This section describes the data required for both the 3-D and 2-D versions of Program Phashift.

3.7.1 2-D Version: One card is required to control the input of data. This card specifies:

Parameter	Format	Columns	Description
EZERO	F10.5	1-10	The smallest energy value desired in the calculation of the phase-shift vs energy plot.
EPSILON	F10.5	11-20	The energy increment added to EZERO to generate the next energy data point.
VULCAN	F10.5	21-30	The absolute value of the well depth.
FERMI	F10.5	31-40	The well width.
MAX	I1	45	The highest order phase shift desired. For S,P,D,F,G-waves, MAX = 1,2,3,4,5.
JELLO	I1	50	Controls units of phase shift. If JELLO is 1, phase shift is expressed in degrees, if JELLO is blank, phase shift is calculated in radians.
LAST	I4	55-58	The total number of data points desired to be calculated, maximum 1000. The largest energy value calculated is given by the product EPSILON X LAST.
NO PLOT	I1	65	If a plot is not desired, enter a 1. Otherwise leave blank.
NOSCALE	I1	70	If set = 1, ordinal scale is 0-1. If blank, automatic self-scaling obtains.
KILL	I1	72	Zero except in last data card of set

3.7.2 3-D Version: Two-plus cards are required to control the input of data.

A maximum of 10 data cards per set are possible. The first card specifies:

Parameter	Format	Columns	Description
EZERO	F10.7	1-10	The smallest energy value desired in the calculation of the phase-shift vs energy plot.
EPSILON	F10.7	11-10	The energy increment added to EZERO to generate the next energy data point.
FERMI	F10.7	21-30	The well width.
MAX	I1	45	The highest order phase shift desired. For S,P,D,F,G-waves, MAX = 1,2,3,4,5.
JELLO	I1	50	Controls units of phase shift. If set to 1, units are degrees, if blank, units are radians.
LAST	I4	55-58	The total number of data points desired to be calculated, maximum 1000. The largest energy value calculated is given by product EPSILON X LAST.
NCURVES	I1	60	The number of curves of different well depth desired to be plotted (maximum of 9).
NO PLOT	I1	65	If a plot is not desired, enter a 1, otherwise leave blank.
NOSCALE	I1	70	If set 1, ordinal scale is 0-1. If blank, automatic self-scaling obtains.

The second and remaining cards specify:

VULCAN	F10.5	1-10	The absolute value of the well depth. Successive data cards generally contain successively larger well depths, up to a maximum of 9 total data cards giving values for VULCAN.
KILL	I1	21	Zero, except on the last data card of each successive data set.

### 3.8 Output

### 3.8.1 Printed output: Both versions

The following information is printed before the data list:

(1) PHASE SHIFT IN RADIANS (The word DEGREES will appear instead of RADIANS if that option is selected) FOR A QUANTUM MECHANICAL PARTICLE SCATTERING OFF A 3-D SQUARE WELL.

(2) Information regarding the chosen input parameters is printed as follows:

WELL DEPTH =  
WELL WIDTH =  
EZERO : The minimum and maximum values along the energy axis.  
ELAST :

(3) Data list: The data list contains 10 variables as follows:

ENERGY (I): The energy of the Ith data point

DELTA (0): The zero order phase shift (S-wave)

DELTA (1): The first order phase shift (P-wave)

DELTA (2): The second order phase shift (D-wave)

DELTA (3): The third order phase shift (F-wave)

DELTA (4): The fourth order phase shift (G-wave)

K : ZEUS(1), the product of the square root of the ENERGY(I) and FERMI (the well width)

ERGO : The product  $K \times FERMI$  (this is what the actual abscissal value is in the plots)

OLYMPUS 1: The numerator of the Arctan function for 1st data point (printed for diagnostic purposes in checking the accuracy of the phase shift calculation) see sect. 4.0.

HADES 1 : The denominator of the Arctan function described above.

### 3.8.2 Plotted Output

2-D version: Samples of the plotted output are given in figures 4, 5, and 6. Each of these figures shows the S, P, D, F, and G-wave phase shifts superimposed on the same plot for comparative purposes. The ordinal values give the phase shifts in units of  $\pi$ , while the abscissal values are products of energy

x well width<sup>2</sup>. The curves shown have been embellished for illustrative purposes, the actual curves are displayed as solid lines, and must be identified individually either by plotting consecutive displays working up from S to G-waves, or by recourse to the printed output. The interpretation of the broken curves displayed in figure 6 is discussed in section 8.0 (REMARKS); it is a result of the indistinguishability of the multiples of  $\pi$  to the computer.

3-D version: A sample of the plotted output is given in figure 3. It is embellished only by the numerals identifying V-values near each curve, these are not displayed on the CalComp plot, otherwise the figure is essentially unretouched. It should be noted that it is not necessary to have sequential V-values; it should also be noted that the maximum point of each individual V curve is displayed between 0 and 1 in units of  $\pi$ , and in no case is greater than 1. In cases where the curve would have penetrated the region between 1 and 2, the computer displays the peak of the curve reduced to the interval 0 to 1, as in curve V = 20.

### 3.9 Formats

In the 2-D version, the data are printed in F10.5 format; in the 3-D version, the data are printed in F11.7 format.

### 3.10 External Routines and Symbols

Both versions:	PHASRPT calls	SPHIN SPHYN MINIMAX LAYOUT STOPPLOT	} -PHASRPT
		ATANK SQRTV	
			} -SYSTEM LIBRARY

### 3.11 Timing

2-D version: To produce figures such as figures 4, 5 and 6, i.e., for 1000 data points and calculations of S thru G wave phase shifts, the processing time is generally less than 1.5 minutes. The plotting time generally runs about 4 minutes.

3-D Version: To produce a figure such as figure 3 generally takes about 4 minutes of computer time plus about 4 minutes of CalComp time.

### 3.12 Accuracy

The accuracy of the calculations was verified using an electronic calculator for several points on the S-wave curves. The accuracy of the plots was verified by comparison with curves for S, P, and D-wave phase shifts presented for  $V = 57.5$  over an  $Ea^2$  interval of 0 to 10, which were presented in Mott and Massey<sup>1</sup>.

### 3.13 Cautions to Users

Experience has shown that the main sources of error in using these routines lies in the confusion which may arise in identifying the individual phase shift curves on the Calcomp plot, especially at the high energy end of the curves.

A second source of confusion may arise from the display of broken curves in figures 3 and 6 due to the compression by the computer of multiples of  $\pi$  into the first "zone" of zero to  $\pi$ . This reduction is demonstrated in figures 2a and 2b, the unreduced and reduced phase shifts as applied to figure 5a, page 33, in Mott and Massey. A method of determining the ordinal intercept, so as to be able to apply Levinson's theorem, which relates the ordinal intercept to the number of bound states the well may support, is discussed in section 8.0, REMARKS.

### 3.14 Program Deck Structure

2-D version: 7JOB card

7EQUIP, 13 = PL

7FTN card

Program Subroutines: PHASHIFT  
SPHJN  
SPHYN  
MINIMAX  
LAYOUT (with COMMON blocks and DIMENSION statements  
SCOPE subscripted 5)

7LOAD card

7RUN card

data cards (see sect. 3.7.1) each data card will generate a series of superimposed phase shift curves for a given well depth; each successive data card will generate another plot.

EØF

3-D version: 7JOB card

7DEMAND, 53776B

7EQUIP, 13 = PL

7FTN

Program subroutines: PHASHIFT  
SPHJN  
SPHYN  
MINIMAX  
LAYOUT (with COMMON blocks and DIMENSION statements  
SCOPE subscripted 9)

7BANK, (0<sup>1</sup>,/3/)

7LOAD card

7RUN card

data cards (see section 3.7.2)

EØF

### 3.15 References - Literature - Appendices

- (1) Mott, N. F. & Massey, H. S. W., THE THEORY OF ATOMIC COLLISIONS, 3rd Ed., Oxford, Clarendon Press, 1965.
- (2) Meijer, P. H. E. & Repace, J. L., "Phase Shifts of the 3-Dimensional Spherically Symmetric Square Well Potential", American Journal of Physics, Vol. 43, No. 5, May 1975, p. 428.
- (3) Newton, R. G., SCATTERING THEORY OF WAVES AND PARTICLES, McGraw-Hill, N.Y., 1968.

### 4.0 METHOD OR ALGORITHM

I. A detailed description of the significance of certain features of the phase shift curves, as well as a guide to the literature is available in reference



(2). The mathematical expression for the phase shifts and simplifying assumptions regarding the constants are discussed below.

#### 4.1 Description of the Phase Shift Equations

We now consider the ideas of the partial wave method. Suppose we have a solution inside a spherical well of radius  $a$  and depth  $V$ . See figure 1. This solution has to match the wavefunctions outside the well at  $r = a$ . At this point the wavefunction should have the same value (continuity of the wavefunction) and the same derivative (continuity of the derivative of the wave function). The continuity of the derivative of the wavefunction is required because the momentum of the particle should be defined at every point in space. There are two more boundary conditions: 1) the wavefunction inside the sphere should be zero at the origin, and 2) the wavefunction outside the sphere should have a fixed amplitude. The value of this amplitude is of no importance when we calculate the phase shift. The first boundary condition excludes Neumann functions inside the sphere. As a result of the spherical symmetry of the potential well, the angular dependence inside and outside should be identical: the  $\delta$ -value outside the sphere.

We introduce the following notation.  $k = \sqrt{E}$ . We use units  $\hbar = 1$  and  $\mu = 1$ .  $\kappa = \sqrt{V+E}$  where  $V$  is the well depth, the absolute value of the negative potential inside the sphere. We use the standard notation for the spherical Bessel and Neumann functions (compare NBS handbook)<sup>10</sup>. The radius of the sphere is  $r = a$ . With this notation we have, for the radial part of the wavefunction inside the sphere:

$$r < a: \psi(r) = A j_l(\kappa r)$$

and for the wavefunction outside the sphere:

$$r > a: \psi(r) = B j_l(kr) + C n_l(kr), \quad B^2 + C^2 = 1$$

The convention is normally to replace the two constants by the sine and cosine

of an angle as follows.

Having established these two wavefunctions, the matching condition gives the following relation:

$$j_L(za) = \cos \delta_L j_L(ka) - \sin \delta_L n_L(ka)$$

$$2j'_L(za) = k (\cos \delta_L j'_L(ka) - \sin \delta_L n'_L(ka))$$

We take the ratio between these two equations, omitting the subscript

$$\frac{j'_L(za)}{j_L(za)} = \frac{k j'_L(ka) - k \tan \delta_L n'_L(ka)}{j_L(ka) - \tan \delta_L n_L(ka)}$$

and solve for  $\tan \delta_L$ . The result can be written as

$$\delta_L = \tan^{-1} \frac{ka j'_L(ka) - \gamma j_L(ka)}{ka n'_L(ka) - \gamma n_L(ka)}$$

where  $\gamma$  is given by

$$\gamma = za j'_L(za)/j_L(za)$$

These are the equations of interest which are used to compute the phase shift  $\delta_L$ .

5.0 Listing of program -- see pages 23 et. seq.

6.0 COMPARISONS - see section 3.12

7.0 TEST METHOD AND RESULTS

2-D version: To generate figures 4, 5 and 6 during one computer run, set

1st data card	2nd data card	3rd data card
EZERO = .00001	.00001	.00001
EPSILON = 0.5	0.5	0.5
VULCAN = 1.0	10.0	100.0
FERMI = 1.0	1.0	1.0
MAX = 5	5	5
JELLO = b	b	b
LAST = 100	100	100
KAPLOT = b	b	b
NOSCALE = 1	1	1
KILL = 0	0	1

The partial printed output from the second data card follows: (See Figure A)

3-D version: To generate figure 3, set

1st data card

EZERO .0000001  
EPSILON .014  
FEMT 1.  
MAX 5  
NULO 6  
LAST 1000  
NOURVES 9  
NOTLOT 6  
NOCASE 1

1st data card	VULCAN	1	KILL	6
2nd " "	"	2	"	6
3rd " "	"	3	"	6
4th " "	"	4	"	6
5th " "	"	10	"	6
6th " "	"	18	"	6
7th " "	"	19	"	6
8th " "	"	20	"	6
10th " "	"	21	"	1

A partial listing of the printed output for figure 3 follows: (see Figure B)

1.1 - CHARTS

Note that the phase shifts are defined only to within an integral multiple of  $\pi$ . It is customary to define  $\lim_{E \rightarrow 0} \delta_l = 0$ . Since the computer delivers values of phase shift only between 0 and 1 in units of  $\pi$  radians, the question arises as to how to interpret these plots in light of Levinson's theorem. According to Levinson's theorem, the value of the phase shift at zero energy, in units of  $\pi$  radians, yields the number of  $l$ -wave bound states that the well can support, except at certain transitional strengths<sup>3</sup>. If multiples of  $\pi$  are suppressed in the plot, how can one determine the number of bound states? It will be noted in figure 6 for the deep well, that there are a number of discontinuities in phase. This is a result of the suppression of multiples of  $n\pi$ . If one plots figure 6 out to energies greater than 400, it will be found that only one more set of discontinuities occurs before the phase shift curves asymptotically approach zero. If one assigns a negative value to jumps up, and a positive value to jumps down, the algebraic number of phase jumps will be equal to the number of  $l$ -wave bound states

PHASE SHIFTS IN RADIAN FOR A QUANTUM MECHANICAL PARTICLE SCATTERING OFF A 3-D SQUARE WELL.

WELL DEPTH = 10.00000 WELL WIDTH = 1.00000 EZERO = 0.00001 ELAST = 50.00000

ENERGY(1)	DELTA(0)	DELTA(1)	DELTA(2)	DELTA(3)	DELTA(4)	K	ER00	OLYMPUS	MAFES1
0.00001	0.99900	1.00000	0.00000	0.00000	0.00000	0.00316	0.00001	-151.84410	*****
0.00001	0.78110	0.86235	0.00000	0.00000	0.00000	0.70711	0.50001	-29.27965	35.84345
1.00001	0.69864	0.79799	0.00374	0.00004	0.00008	1.00000	1.00001	-15.24113	10.97450
1.50001	0.63937	0.75005	0.01042	0.00016	0.00000	1.22475	1.50001	-9.87993	4.62528
2.00001	0.59300	0.71044	0.02112	0.00043	0.00001	1.41422	2.00001	-7.08427	2.11079
2.50001	0.55521	0.67615	0.03585	0.00091	0.00002	1.58114	2.50001	-5.58946	0.94479
3.00001	0.52376	0.64566	0.05208	0.00168	0.00004	1.73285	3.00001	-4.24666	0.31909
3.50001	0.49731	0.61807	0.06926	0.00279	0.00005	1.87083	3.50001	-3.47843	-0.02918
4.00001	0.47499	0.59282	0.12895	0.00433	0.00015	2.00000	4.00001	-2.92234	-0.22844
4.50001	0.45617	0.56350	0.15991	0.00636	0.00025	2.12132	4.50001	-2.46659	-0.34711
5.00001	0.44000	0.53783	0.21205	0.00895	0.00039	2.23007	5.00001	-2.13521	-0.44640
5.50001	0.42732	0.52760	0.25191	0.01216	0.00058	2.34521	5.50001	-1.87552	-0.43593
6.00001	0.41662	0.52065	0.27698	0.01606	0.00043	2.44949	6.00001	-1.64996	-0.44772
6.50001	0.40803	0.49803	0.31573	0.02071	0.00115	2.54951	6.50001	-1.50626	-0.43771
7.00001	0.41131	0.47407	0.33842	0.02616	0.00155	2.64575	7.00001	-1.37523	-0.44057
7.50001	0.39622	0.45827	0.35556	0.03245	0.00205	2.73861	7.50001	-1.27015	-0.42943
8.00001	0.39253	0.44335	0.36809	0.03960	0.00285	2.82843	8.00001	-1.18595	-0.41637
8.50001	0.39000	0.43927	0.37681	0.04763	0.00337	2.91548	8.50001	-1.11878	-0.40279
9.00001	0.38842	0.43597	0.38249	0.05653	0.00422	3.00000	9.00001	-1.06505	-0.38944
9.50001	0.38757	0.43461	0.38576	0.06626	0.00522	3.08221	9.50001	-1.02421	-0.37749
10.00001	0.38725	0.43417	0.38713	0.07677	0.00637	3.16228	10.00001	-0.99242	-0.36747
10.50001	0.38728	0.43400	0.38698	0.08796	0.00749	3.24037	10.50001	-0.96938	-0.35839
11.00001	0.38748	0.43389	0.38693	0.09973	0.00819	3.31693	11.00001	-0.94931	-0.35178
11.50001	0.38772	0.43381	0.38633	0.11193	0.01048	3.39117	11.50001	-0.93434	-0.34712
12.00001	0.38774	0.43374	0.38528	0.12440	0.01277	3.46117	12.00001	-0.92091	-0.34517
12.50001	0.38785	0.43267	0.37608	0.13698	0.01498	3.53554	12.50001	-0.91919	-0.34510
13.00001	0.38756	0.43197	0.37246	0.14948	0.01721	3.60555	13.00001	-0.94407	-0.34840
13.50001	0.38722	0.43143	0.36794	0.16175	0.01977	3.67424	13.50001	-0.95227	-0.35309
14.00001	0.38613	0.43144	0.36313	0.17362	0.02257	3.74166	14.00001	-0.96486	-0.35670
14.50001	0.38489	0.43128	0.35810	0.18495	0.02541	3.80749	14.50001	-0.98048	-0.37047
15.00001	0.38334	0.43070	0.35290	0.19565	0.02891	3.87298	15.00001	-0.99931	-0.38348
15.50001	0.38119	0.43161	0.34758	0.20563	0.03245	3.93701	15.50001	-1.02110	-0.39922
16.00001	0.37914	0.43256	0.34218	0.21483	0.03624	4.00000	16.00001	-1.04641	-0.41747
16.50001	0.37658	0.43220	0.33623	0.22323	0.04029	4.06502	16.50001	-1.07469	-0.43890
17.00001	0.37374	0.43177	0.33117	0.23060	0.04458	4.12311	17.00001	-1.10623	-0.46334
17.50001	0.37064	0.43143	0.32591	0.23757	0.04910	4.18330	17.50001	-1.14121	-0.49113
18.00001	0.36713	0.43150	0.32038	0.24354	0.05346	4.24264	18.00001	-1.17988	-0.52251
18.50001	0.36372	0.43149	0.31499	0.24876	0.05843	4.30116	18.50001	-1.22227	-0.55740
19.00001	0.36031	0.43142	0.30966	0.25325	0.06400	4.35890	19.00001	-1.26969	-0.59740
19.50001	0.35612	0.43144	0.30441	0.25787	0.06916	4.41588	19.50001	-1.32110	-0.64140
20.00001	0.35208	0.43140	0.29924	0.26266	0.07487	4.47214	20.00001	-1.37698	-0.68140
20.50001	0.34793	0.43140	0.29417	0.26767	0.08093	4.52769	20.50001	-1.43899	-0.72755
21.00001	0.34359	0.43140	0.28921	0.27263	0.08730	4.58258	21.00001	-1.50840	-0.78104
21.50001	0.33937	0.43140	0.28436	0.27763	0.09416	4.63681	21.50001	-1.58495	-0.84105
22.00001	0.33500	0.43140	0.27943	0.28261	0.09888	4.69042	22.00001	-1.66851	-0.91314
22.50001	0.33059	0.43140	0.27503	0.28761	0.10403	4.74342	22.50001	-1.76031	-1.00514
23.00001	0.32614	0.43140	0.27056	0.29263	0.10998	4.79583	23.00001	-1.86121	-1.11659
23.50001	0.32170	0.43140	0.26624	0.29763	0.11590	4.84768	23.50001	-1.97255	-1.24899
24.00001	0.31725	0.43140	0.26204	0.30263	0.12176	4.89898	24.00001	-2.09525	-1.40714
24.50001	0.31283	0.43140	0.25800	0.30763	0.12754	4.94975	24.50001	-2.22106	-1.43714
25.00001	0.30843	0.43140	0.25412	0.31263	0.13320	5.00000	25.00001	-2.35001	-1.61440
25.50001	0.30404	0.43140	0.25038	0.31763	0.13875	5.04975	25.50001	-2.48271	-1.83175
26.00001	0.29976	0.43140	0.24690	0.32263	0.14419	5.09821	26.00001	-2.61996	-2.10272
26.50001	0.29551	0.43140	0.24338	0.32763	0.14929	5.14782	26.50001	-2.76201	-2.45049
27.00001	0.29132	0.43140	0.24012	0.33263	0.15429	5.19615	27.00001	-2.90941	-2.81508
27.50001	0.28710	0.43140	0.23690	0.33763	0.15914	5.24399	27.50001	-3.06241	-3.18445

PHASE SHIFTS IN RADIANS FOR A QUANTUM MECHANICAL PARTICLE SCATTERING OFF A 3-D SQUARE WELL.

WELL DEPTH = 1.00000 WELL WIDTH = 1.000000 EZERO = 0.000001 ELAST = 1.0000000000

ENERGY	DELTA(0)	DELTA(1)	DELTA(2)	DELTA(3)	DELTA(4)	ERG0	OLYMPUS	WDFSI
0.000001	0.0000561	0.0000000	0.0000000	0.0000000	0.0000000	0.0000001	0.0000001	0.0000000
0.0140001	0.0208237	0.0000406	0.0000000	0.0000000	0.0000000	0.0183220	0.0000001	0.0000000
0.0280001	0.0292128	0.0001148	0.0000000	0.0000000	0.0000000	0.0273123	0.0000001	0.0000000
0.0420001	0.0354937	0.0002106	0.0000000	0.0000000	0.0000000	0.0348882	0.0000001	0.0000000
0.0560001	0.0406613	0.0003237	0.0000000	0.0000000	0.0000000	0.0397800	0.0000001	0.0000000
0.0700001	0.0451052	0.0004517	0.0000000	0.0000000	0.0000000	0.0437424	0.0000001	0.0000000
0.0840001	0.0490272	0.0005929	0.0000000	0.0000000	0.0000000	0.0473661	0.0000001	0.0000000
0.0980001	0.0525481	0.0007460	0.0000000	0.0000000	0.0000000	0.0509877	0.0000001	0.0000000
0.1120001	0.0557477	0.0009160	0.0000000	0.0000000	0.0000000	0.0546147	0.0000001	0.0000000
0.1260001	0.0586919	0.0010842	0.0000000	0.0000000	0.0000000	0.0582462	0.0000001	0.0000000
0.1400001	0.0613917	0.0012679	0.0000000	0.0000000	0.0000000	0.0618840	0.0000001	0.0000000
0.1540001	0.0639024	0.0014605	0.0000000	0.0000000	0.0000000	0.0654459	0.0000001	0.0000000
0.1680001	0.0662566	0.0016616	0.0000000	0.0000000	0.0000000	0.0689782	0.0000001	0.0000000
0.1820001	0.0684560	0.0018707	0.0000000	0.0000000	0.0000000	0.0724817	0.0000001	0.0000000
0.1960001	0.0705227	0.0020874	0.0000000	0.0000000	0.0000000	0.0759495	0.0000001	0.0000000
0.2100001	0.0724701	0.0023114	0.0000000	0.0000000	0.0000000	0.0793844	0.0000001	0.0000000
0.2240001	0.0743095	0.0025424	0.0000000	0.0000000	0.0000000	0.0827890	0.0000001	0.0000000
0.2380001	0.0760505	0.0027801	0.0000000	0.0000000	0.0000000	0.0861691	0.0000001	0.0000000
0.2520001	0.0777015	0.0030242	0.0000000	0.0000000	0.0000000	0.0895254	0.0000001	0.0000000
0.2660001	0.0792697	0.0032746	0.0000000	0.0000000	0.0000000	0.0928982	0.0000001	0.0000000
0.2800001	0.0807613	0.0035309	0.0000000	0.0000000	0.0000000	0.0962871	0.0000001	0.0000000
0.2940001	0.0821820	0.0037930	0.0000000	0.0000000	0.0000000	0.0996921	0.0000001	0.0000000
0.3080001	0.0835367	0.0040607	0.0000000	0.0000000	0.0000000	0.1031134	0.0000001	0.0000000
0.3220001	0.0848298	0.0043338	0.0000000	0.0000000	0.0000000	0.1065505	0.0000001	0.0000000
0.3360001	0.0860654	0.0046122	0.0000000	0.0000000	0.0000000	0.1100034	0.0000001	0.0000000
0.3500001	0.0872469	0.0048955	0.0000000	0.0000000	0.0000000	0.1134721	0.0000001	0.0000000
0.3640001	0.0883777	0.0051840	0.0000000	0.0000000	0.0000000	0.1169564	0.0000001	0.0000000
0.3780001	0.0894607	0.0054772	0.0000000	0.0000000	0.0000000	0.1204561	0.0000001	0.0000000
0.3920001	0.0904986	0.0057750	0.0000000	0.0000000	0.0000000	0.1239712	0.0000001	0.0000000
0.4060001	0.0914939	0.0060773	0.0000000	0.0000000	0.0000000	0.1275017	0.0000001	0.0000000
0.4200001	0.0924488	0.0063840	0.0000000	0.0000000	0.0000000	0.1310477	0.0000001	0.0000000
0.4340001	0.0933655	0.0066951	0.0000000	0.0000000	0.0000000	0.1346091	0.0000001	0.0000000
0.4480001	0.0942458	0.0070102	0.0000000	0.0000000	0.0000000	0.1381859	0.0000001	0.0000000
0.4620001	0.0950915	0.0073295	0.0000000	0.0000000	0.0000000	0.1417781	0.0000001	0.0000000
0.4760001	0.0959044	0.0076527	0.0000000	0.0000000	0.0000000	0.1453856	0.0000001	0.0000000
0.4900001	0.0966860	0.0079797	0.0000000	0.0000000	0.0000000	0.1489984	0.0000001	0.0000000
0.5040001	0.0974376	0.0083106	0.0000000	0.0000000	0.0000000	0.1526264	0.0000001	0.0000000
0.5180001	0.0981607	0.0086450	0.0000000	0.0000000	0.0000000	0.1562694	0.0000001	0.0000000
0.5320001	0.0988564	0.0089831	0.0000000	0.0000000	0.0000000	0.1599273	0.0000001	0.0000000
0.5460001	0.0995266	0.0093246	0.0000000	0.0000000	0.0000000	0.1635999	0.0000001	0.0000000
0.5600001	0.1001767	0.0096696	0.0000000	0.0000000	0.0000000	0.1672871	0.0000001	0.0000000
0.5740001	0.1008113	0.0100179	0.0000000	0.0000000	0.0000000	0.1709888	0.0000001	0.0000000
0.5880001	0.1014289	0.0103694	0.0000000	0.0000000	0.0000000	0.1747050	0.0000001	0.0000000
0.6020001	0.1020264	0.0107241	0.0000000	0.0000000	0.0000000	0.1784366	0.0000001	0.0000000
0.6160001	0.1026044	0.0110819	0.0000000	0.0000000	0.0000000	0.1821836	0.0000001	0.0000000
0.6300001	0.1031687	0.0114477	0.0000000	0.0000000	0.0000000	0.1859460	0.0000001	0.0000000
0.6440001	0.1037226	0.0118204	0.0000000	0.0000000	0.0000000	0.1897237	0.0000001	0.0000000
0.6580001	0.1042689	0.0121950	0.0000000	0.0000000	0.0000000	0.1935168	0.0000001	0.0000000
0.6720001	0.1048061	0.0125745	0.0000000	0.0000000	0.0000000	0.1973259	0.0000001	0.0000000
0.6860001	0.1053359	0.0129587	0.0000000	0.0000000	0.0000000	0.2011500	0.0000001	0.0000000
0.7000001	0.1058579	0.0133486	0.0000000	0.0000000	0.0000000	0.2049899	0.0000001	0.0000000
0.7140001	0.1063724	0.0137437	0.0000000	0.0000000	0.0000000	0.2088456	0.0000001	0.0000000
0.7280001	0.1068791	0.0141442	0.0000000	0.0000000	0.0000000	0.2127171	0.0000001	0.0000000
0.7420001	0.1073782	0.0145499	0.0000000	0.0000000	0.0000000	0.2166044	0.0000001	0.0000000
0.7560001	0.1078698	0.0149604	0.0000000	0.0000000	0.0000000	0.2205075	0.0000001	0.0000000
0.7700001	0.1083541	0.0153757	0.0000000	0.0000000	0.0000000	0.2244263	0.0000001	0.0000000
0.7840001	0.1088311	0.0157964	0.0000000	0.0000000	0.0000000	0.2283607	0.0000001	0.0000000
0.7980001	0.1093008	0.0162224	0.0000000	0.0000000	0.0000000	0.2323106	0.0000001	0.0000000
0.8120001	0.1097631	0.0166537	0.0000000	0.0000000	0.0000000	0.2362759	0.0000001	0.0000000
0.8260001	0.1102189	0.0170894	0.0000000	0.0000000	0.0000000	0.2402572	0.0000001	0.0000000
0.8400001	0.1106682	0.0175294	0.0000000	0.0000000	0.0000000	0.2442544	0.0000001	0.0000000
0.8540001	0.1111117	0.0179737	0.0000000	0.0000000	0.0000000	0.2482674	0.0000001	0.0000000
0.8680001	0.1115486	0.0184224	0.0000000	0.0000000	0.0000000	0.2522961	0.0000001	0.0000000
0.8820001	0.1119799	0.0188754	0.0000000	0.0000000	0.0000000	0.2563403	0.0000001	0.0000000
0.8960001	0.1124056	0.0193319	0.0000000	0.0000000	0.0000000	0.2603999	0.0000001	0.0000000
0.9100001	0.1128257	0.0197920	0.0000000	0.0000000	0.0000000	0.2644749	0.0000001	0.0000000
0.9240001	0.1132402	0.0202564	0.0000000	0.0000000	0.0000000	0.2685652	0.0000001	0.0000000
0.9380001	0.1136491	0.0207242	0.0000000	0.0000000	0.0000000	0.2726717	0.0000001	0.0000000
0.9520001	0.1140524	0.0211954	0.0000000	0.0000000	0.0000000	0.2767944	0.0000001	0.0000000
0.9660001	0.1144501	0.0216699	0.0000000	0.0000000	0.0000000	0.2809332	0.0000001	0.0000000
0.9800001	0.1148422	0.0221477	0.0000000	0.0000000	0.0000000	0.2850880	0.0000001	0.0000000
0.9940001	0.1152287	0.0226289	0.0000000	0.0000000	0.0000000	0.2892587	0.0000001	0.0000000
1.0080001	0.1156098	0.0231134	0.0000000	0.0000000	0.0000000	0.2934452	0.0000001	0.0000000
1.0220001	0.1159859	0.0235999	0.0000000	0.0000000	0.0000000	0.2976475	0.0000001	0.0000000
1.0360001	0.1163570	0.0240888	0.0000000	0.0000000	0.0000000	0.3018656	0.0000001	0.0000000
1.0500001	0.1167231	0.0245799	0.0000000	0.0000000	0.0000000	0.3060994	0.0000001	0.0000000
1.0640001	0.1170842	0.0250731	0.0000000	0.0000000	0.0000000	0.3103489	0.0000001	0.0000000
1.0780001	0.1174403	0.0255684	0.0000000	0.0000000	0.0000000	0.3146140	0.0000001	0.0000000
1.0920001	0.1177914	0.0260659	0.0000000	0.0000000	0.0000000	0.3188946	0.0000001	0.0000000
1.1060001	0.1181375	0.0265654	0.0000000	0.0000000	0.0000000	0.3231907	0.0000001	0.0000000
1.1200001	0.1184786	0.0270670	0.0000000	0.0000000	0.0000000	0.3275023	0.0000001	0.0000000
1.1340001	0.1188147	0.0275706	0.0000000	0.0000000	0.0000000	0.3318294	0.0000001	0.0000000
1.1480001	0.1191458	0.0280762	0.0000000	0.0000000	0.0000000	0.3361720	0.0000001	0.0000000
1.1620001	0.1194719	0.0285837	0.0000000	0.0000000	0.0000000	0.3405301	0.0000001	0.0000000
1.1760001	0.1197930	0.0290931	0.0000000	0.0000000	0.0000000	0.3449037	0.0000001	0.0000000
1.1900001	0.1201091	0.0296043	0.0000000	0.0000000	0.0000000	0.3492928	0.0000001	0.0000000
1.2040001	0.1204202	0.0301174	0.0000000	0.0000000	0.0000000	0.3536974	0.0000001	0.0000000
1.2180001	0.1207263	0.0306324	0.0000000	0.0000000	0.0000000	0.3581175	0.0000001	0.0000000
1.2320001	0.1210274	0.0311494	0.0000000	0.0000000	0.0000000	0.3625530	0.0000001	0.0000000
1.2460001	0.1213235	0.0316684	0.0000000	0.0000000	0.0000000	0.3670040	0.0000001	0.0000000
1.2600001	0.1216146	0.0321894	0.0000000	0.0000000	0.0000000	0.3714704	0.0000001	0.0000000
1.2740001	0.1219007	0.0327124	0.0000000	0.0000000	0.0000000	0.3759521	0.0000001	0.0000000
1.2880001	0.1221818	0.0332374	0.0000000	0.0				

in the well, for each curve.

#### 9.0 ACKNOWLEDGEMENT

The author is indebted to Professor P. H. E. Meijer, who suggested this work, and to Mr. S. Brenner, who contributed subroutines LAYOUT, SCALE and AXIS, to Mrs. J. Mason, who contributed subroutines SPHJN and SPNTN, and to Dr. A. H. Lowrey, for helpful discussions.

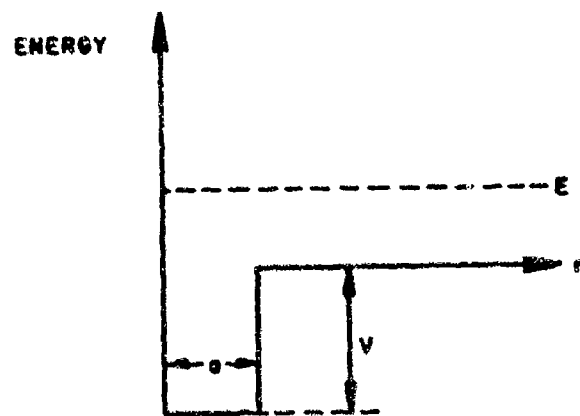


Fig. 1 - The square well potential

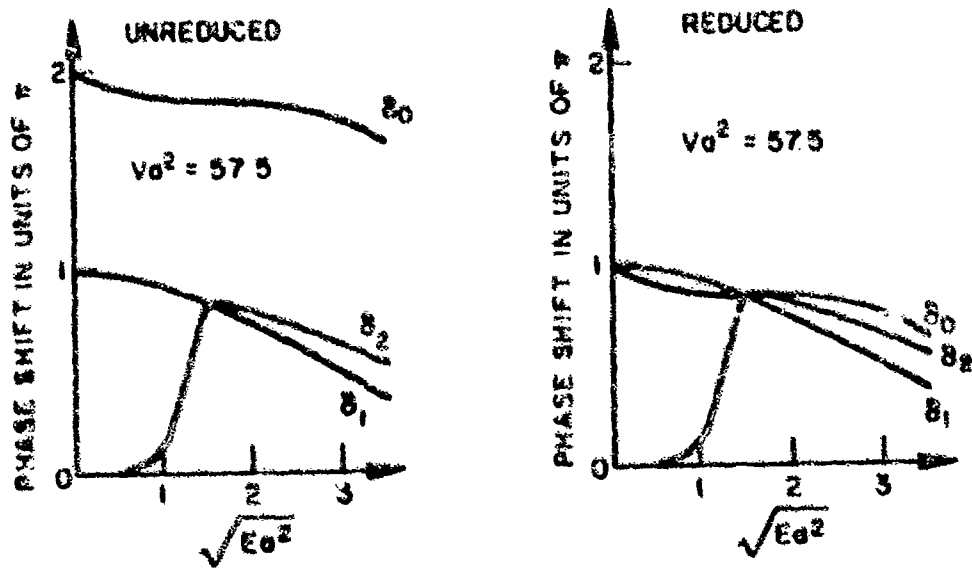


Fig. 2 - Deep well phase shift, unreduced and reduced

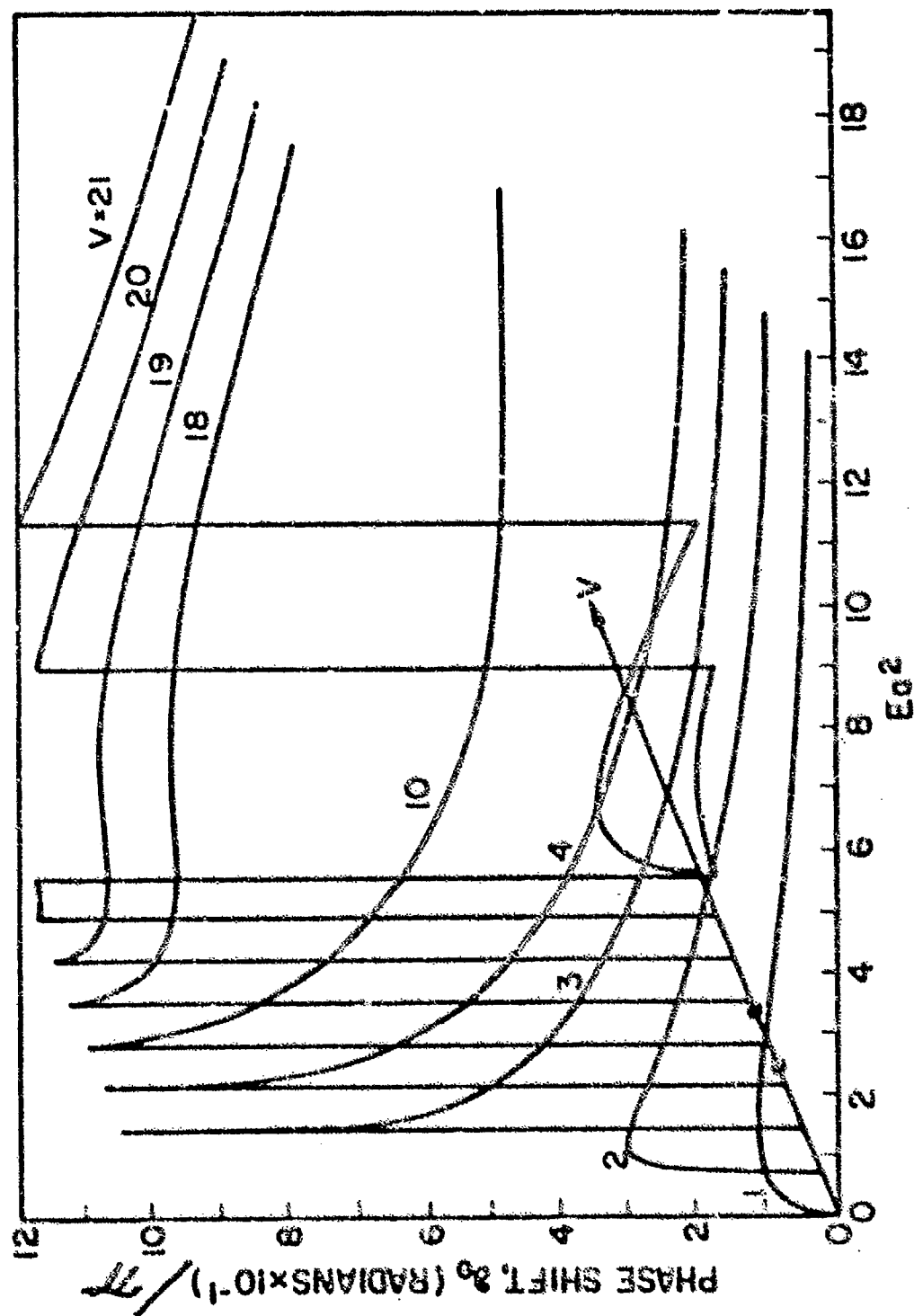


Fig. 3 - Reduced phase shift for  $\ell = 0$  for various values of the well depth  $V$ , in units  $\delta g/\pi$ , where  $\delta g$  is expressed in radians, and where  $a$  is taken as unity



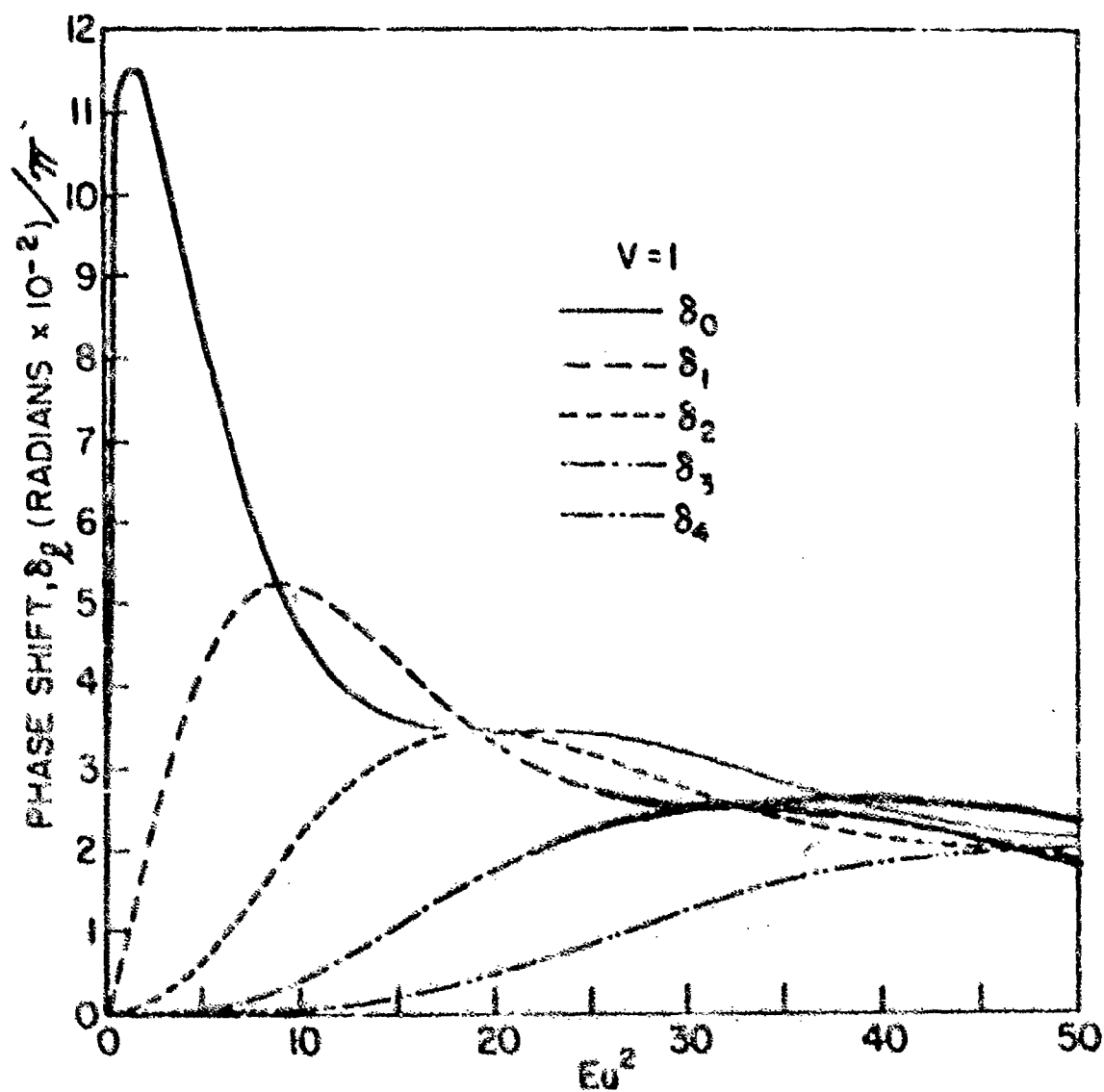


Fig. 4 - Shadow well phase shifts for  $l = 1, 2, 3, 4, 5$  in units  $\delta_l/\pi$ , where  $\delta_l$  is expressed in radians, and where  $\pi$  is taken as unity

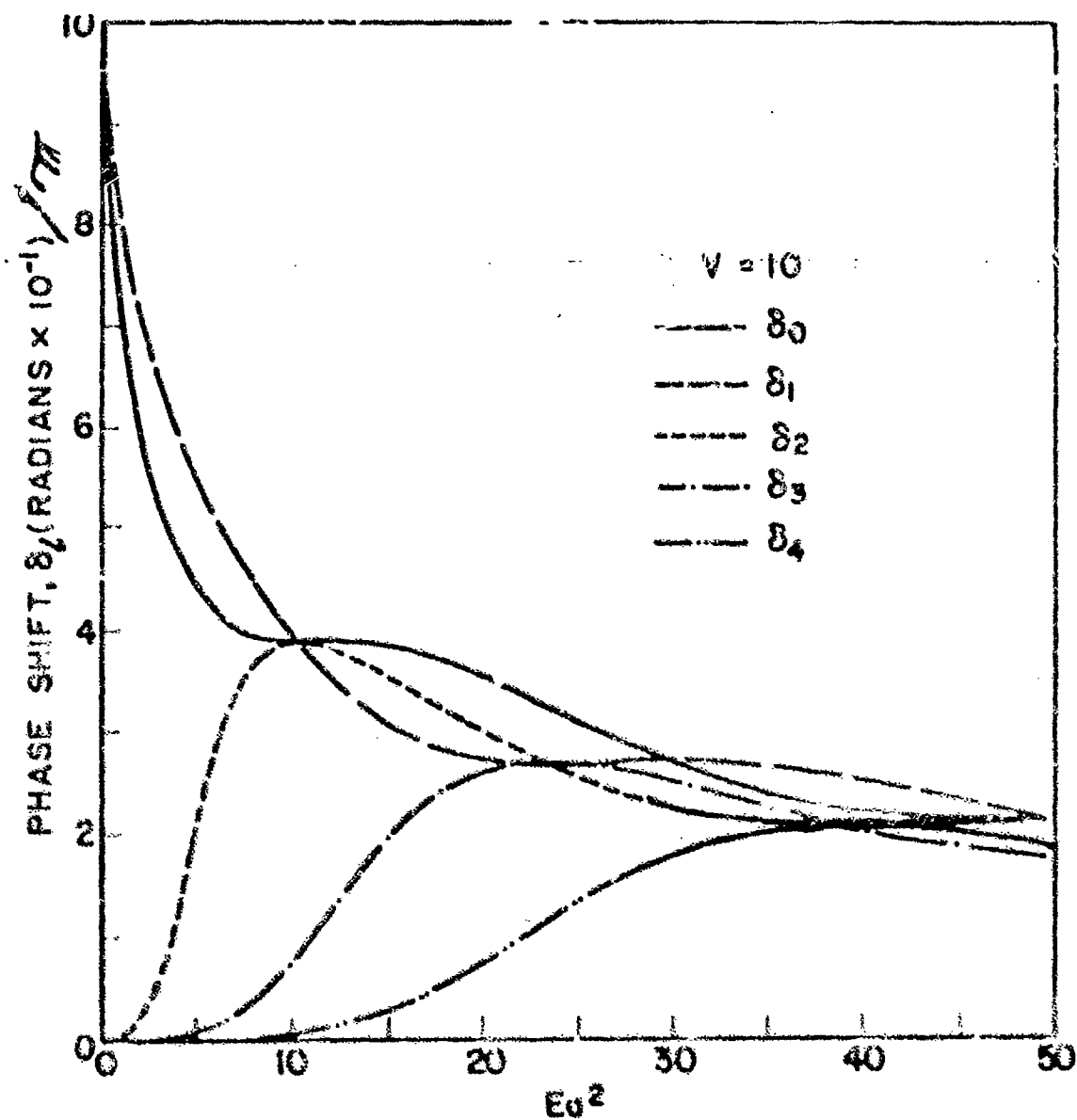


Fig. 6 -- Medium well phase shifts for  $l = 0, 1, 2, 3, 4$  in units  $\delta_l/\pi$ , where  $\delta_l$  is expressed in radians, and where  $\pi$  is taken as unity

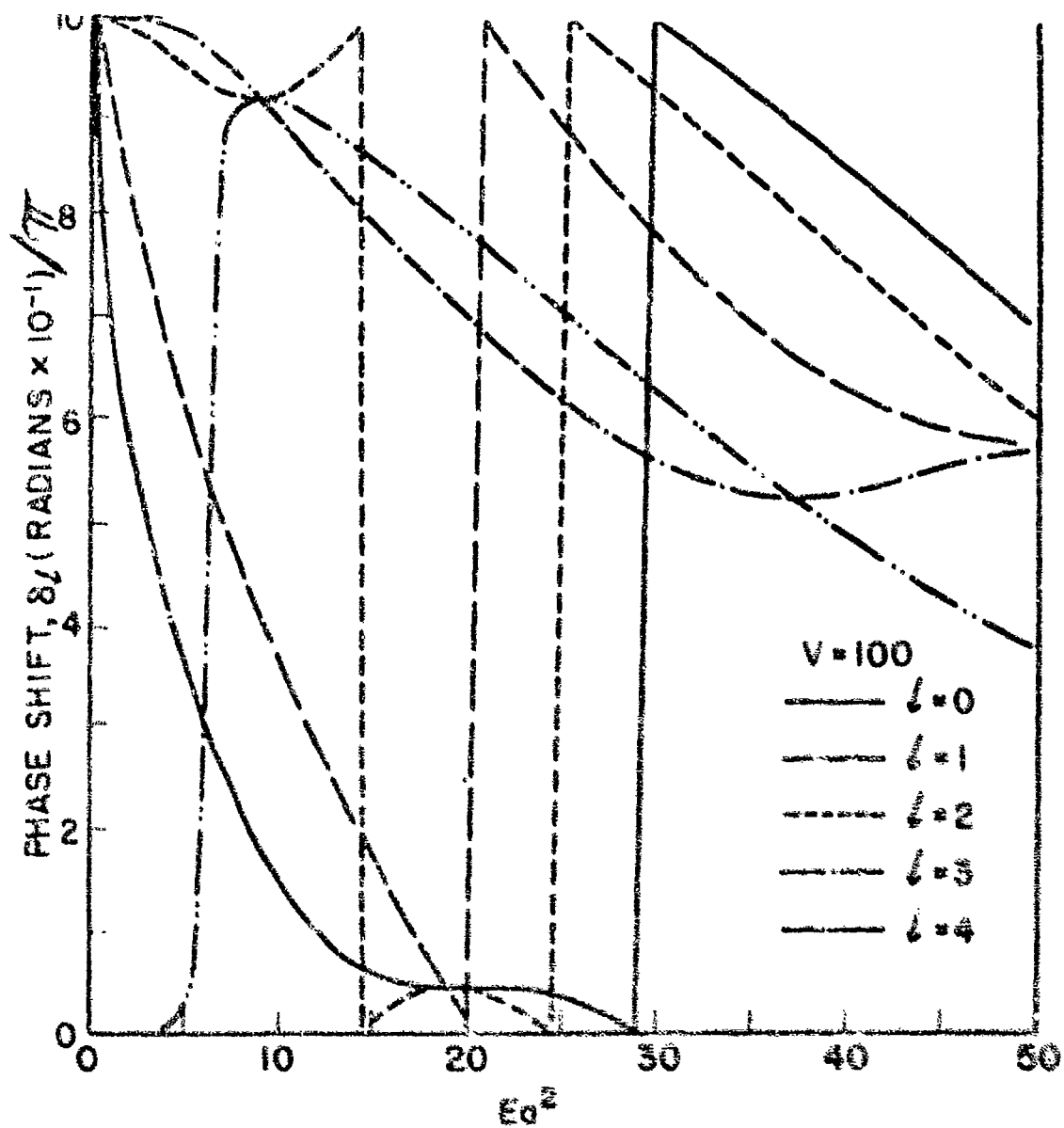


Fig. 6 -- Deep well reduced phase shifts for  $l = 0, 1, 2, 3, 4$  in units  $\delta_0/\pi$ , where  $\delta_0$  is expressed in radians, and where  $\pi$  is taken as unity

## 2-D Version

<pre> PROGRAM PHASHIFT C TWO DIMENSIONAL VERSION C THIS PROGRAM CALCULATES PHASE SHIFTS IN RADIANS OR DEGREES VS E**2 C OR K OR E IN DIMENSIONLESS UNITS FOR THE QUANTUM MECHANICAL SCATTERING C OF A PARTICLE OF ENERGY E FROM A 3-D SQUARE WELL OF WIDTH A AND DEPTH C -V. THE INPUT PARAMETERS ARE ( 1). EZERO (THE LOWEST ENERGY VALUE) CO COLS 1-10. (2). EPSILON (THE ENERGY INCREMENT) COLS 11-20. (3). /VULCAN C THE ABSOLUTE VALUE OF THE WELL DEPTH) COLS 21-30 C (4). FERMI (THE WELL WIDTH) COLS 31-40. (5). MAX (THE C HIGHEST ORDER PHASE SHIFT DESIRED. FOR S,P,D,F) MAX=1,2,3,4. COL 45 C (6). KILL (NONZERO ONLY ON LAST DATA CARD) COL 72. (7). LAST (THE TOT C AL NUMBER OF POINTS DESIRED TO BE CALCULATED). MAX. 1000. COLS 55-58 C (8). JELLO (IF PRINTOUT OF PHASE SHIFTS IN DEGREES IS DESIRED, ENTER C 1 IN COL 50. IF RADIANS DESIRED, LEAVE COL 50 BLANK) COL 50 C IF NO PLOT IS DESIRED, ENTER A 1 IN COL. 65. IF ORDINAL SELF-SCALING C DESIRED, ENTER A 1 IN COL 70, AND INSERT VALUES OF YMIN AND YMAX BEFO C T CALL. COMMON/LABEL/ILX(5),ILY(5),ILH(5),ILR(5),ILC(5),ILABLE(5,5) COMMON /RANGE/ XMAX,XMIN,YMAX,YMIN,IFIRST(5),ILST(5),XSIZE REAL ILX,ILY,ILH,ILR REAL N REAL NZ COMMON/1/ DELTA(1000),CASTOR(1000),POLLUX(1000),HERCULES(1000),AJA 1X(1000),OLYMPUS(1000),HADES(1000),B(1000),DB(1000),N(1000),DN(1000 2),BZ(1000),DRZ(1000),NZ(1000),GAMMA(1000),ENERGY(1000),CYCLOPS(100 30),CENTAUR(1000),ZFUS(1000),RHO(1000),DNZ(1000) DIMENSION DELTA1(1000),DELTA2(1000),DELTA3(1000),DELTA 14(1000),W(5) DIMENSION ERGO(1000) COMMON/A/ PHASE(5000),ERGON(5000) 6 READ (6,10) EZERO,EPSILON,VULCAN,FERMI,MAX,JELLO,LAST,NOPLOT,NOSC 1ALF,KILL 10 FORMAT(4(F10.5),4X,11,4X,11,4X,14,6X,11,4X,11,1X,11) IFIRST(1)=1 IFIRST(2)=1+ LAST IFIRST(3)=1+ 2*LAST IFIRST(4)=1+ 3*LAST IFIRST(5)=1+ 4*LAST ILST(1)= LAST ILST(2)=2*LAST ILST(3)=3*LAST ILST(4)=4*LAST ILST(5)=5*LAST IF (JELLO.EQ.1) GO TO 2 1 PRINT 6 GO TO 3 2 PRINT 7 3 PRINT 12,VULCAN,FERMI,EZERO,XMAX PRINT 8 4 FORMAT (1X,*PHASE SHIFTS IN RADIANS FOR A QUANTUM MECHANICAL PARTI 1CLE SCATTERING OFF A 3-D SQUARE WELL.*/) 7 FORMAT (1X,*PHASE SHIFTS IN DEGREES FOR A QUANTUM MECHANICAL PARTI 1CLE SCATTERING OFF A 3-D SQUARE WELL.*/) 8 FORMAT(5X,*ENERGY(1)*4X*DELTA(1)*4X*DELTA(1)*4X*DELTA(2)*4X*DELTA( 13)*4X*DELTA(4)*4X* K *4X* ERGO *4X*OLYMPUS1*4X*HADES1*/) 12 FORMAT(1X,*WELL DEPTH= *F10.5* WELL WIDTH= *F10.5* EZERO= * 1 F10.5* FLAST= *F10.5/) K=MAX+1 ENERGY(1)=EZERO </pre>	<pre> 100 200 300 400 500 600 700 800 900 1000 1100 1200 1300 1400 1500 1600 1700 1800 1900 2000 2100 2200 2300 2400 2500 2600 2700 2800 2900 3000 3100 3200 3300 3400 3500 3600 3700 3800 3900 4000 4100 4200 4300 4400 4500 4600 4700 4800 4900 5000 5100 5200 5300 5400 5500 5600 5700 5800 5900 </pre>
--	--

DO 91 I=1, LAST	6000
ERGO(I)=ENRGY(I)*(FERMI)**2	6100
14 ENRGY(I+1)=FPSILON+ENRGY(I)	6200
15 CYCLOPS(I)=SORT(ENRGY(I))	6300
20 CENTAUR(I)=SORT(ENRGY(I)+VULCAN)	6400
25 ZEUS(I)=CYCLOPS(I)*FERMI	6500
ZEUSS=ZEUS(I)	6600
27 IF(ZEUSS.EQ.0) 78,30	6700
30 RHO(I)=CENTAUR(I)*FERMI	6800
RHO0=RHO(I)	6900
34 CALL SPHJN(RHO0,K,R)	7000
CALL SPHYN(RHO0,K,N)	7100
DB(I)=-R(I)	7200
DN(I)=-N(I)	7300
DO 35 JI=2, MAX	7400
DR(JI)=(JI*R(JI-1)-(JI+1)*R(JI+1))/(2*JI+1)	7500
DN(JI)=(JI*N(JI-1)-(JI+1)*N(JI+1))/(2*JI+1)	7600
35 CONTINUE	7700
CALL SPHJN(ZEUSS,K,RZ)	7800
CALL SPHYN(ZEUSS,K,NZ)	7900
DRZ(I)=-RZ(I)	8000
DNZ(I)=-NZ(I)	8100
37 DO 38 KI=2, MAX	8200
DBZ(KI)=(KI*RZ(KI-1)-(KI+1)*RZ(KI+1))/(2*KI+1)	8300
DNZ(KI)=(KI*NZ(KI-1)-(KI+1)*NZ(KI+1))/(2*KI+1)	8400
38 CONTINUE	8500
40 DO 76 LI=1, MAX	8600
44 GAMMA(LI)=CENTAUR(I)*DB(LI)/R(LI)	8700
45 CASTOR(LI)=CYCLOPS(I)*DRZ(LI)	8800
50 POLLUX(LI)=GAMMA(LI)*RZ(LI)	8900
55 HERCULES(LI)=CYCLOPS(I)*DNZ(LI)	9000
60 AJAX(LI)=GAMMA(LI)*NZ(LI)	9100
65 OLYMPUS(LI)=CASTOR(LI)-POLLUX(LI)	9200
70 HADES(LI)=HERCULES(LI)-AJAX(LI)	9300
75 W(LI)=ATAN2(OLYMPUS(LI),HADES(LI))	9400
IF(W(LI).LT.0) W(LI)=W(LI)+3.14159	9500
IF(JELLO.EQ.0) GO TO 755	9600
W(LI)=W(LI)*57.29578	9700
IF(JELLO.EQ.1) GO TO 76	9800
75 W(LI)=W(LI)/57.29578	9900
76 CONTINUE	10000
77 GO TO 80	10100
79 DO 79 LI=1, MAX	10200
W(LI)=0.	10300
70 CONTINUE	10400
OLYMPUS(I)=0.	10500
HADES(I)=0.	10600
80 PRINT 85, ENERGY(I), W(1), W(2), W(3), W(4), W(5), ZEUS(I), ERGO(I), OL	10700
YMPUS(I), HADES(I)	10800
85 FORMAT (10(2X,F10.5))	10900
DELTA(I)=W(I)	11000
DELTA1(I)=W(2)	11100
DELTA2(I)=W(3)	11200
DELTA3(I)=W(4)	11300
DELTA4(I)=W(5)	11400
91 CONTINUE	11500
86 PRINT 87	11600
87 FORMAT (///)	11700
IF(NOPLOT.EQ.1) GO TO 95	11800

IF (NOSCALE.EQ.1)GO TO 99	11900
CALL MINIMAX(DELTA,YMIN,YMAX,LAST)	12000
99 XSIZE=10.	12100
XMIN=ERGO(1)	12200
XMAX=ERGO(LAST)	12300
C IF SELF-SCALING NOT USED8 THEN YMIN AND YMAX VALUES MUST BE SUPPLIED.	12400
YMIN=0.	12500
YMAX=1.	12600
C IF DESIRED: DIFFERENT ORDER PHASE SHIFTS MAY BE PLOTTED ON THE SAME.	12700
C GRAPH. USE NOSCALE TO BYPASS. PUT A 1 IN COL 70. SUPPLY YMIN,YMAX.	12800
NLAST=5*LAST	12900
J=LAST	13000
DO 90 M=1,J	13100
PHASE(M)=DELTA(M)	13200
PHASE(M+J)=DELTA1(M)	13300
PHASE(M+2*J)=DELTA2(M)	13400
PHASE(M+3*J)=DELTA3(M)	13500
PHASE(M+4*J)=DELTA4(M)	13600
ERGON(M)=ERGO(M)	13700
ERGON(M+J)=ERGO(M)	13800
ERGON(M+2*J)=ERGO(M)	13900
ERGON(M+3*J)=ERGO(M)	14000
ERGON(M+4*J)=ERGO(M)	14100
90 CONTINUE	14200
CALL LAYOUT(ERGON,PHASE,NLAST,6HE*A**2,7H-DELTA-06,7,5)	14300
PRINT 100	14400
100 FORMAT(///)	14500
95 IF(KILL.EQ.0) GO TO 9	14600
CALL STOPPLOT	14700
190 END	14800
SUBROUTINE MINIMAX(ARRAY,YMIN,YMAX,LAST;	14900
DIMENSION ARRAY(1000)	15000
C THIS SUBROUTINE FINDS BOTH THE LARGEST AND SMALLEST NUMBERS IN ARRAY.	15100
C LARGEST NUMBER	15200
J=1	15300
JJ=J+J	15400
10 DO 30 K=JJ,LAST	15500
IF(ARRAY(J).GE.ARRAY(K))30,20	15600
20 J=K	15700
IF(K.EQ.LAST) GO TO 40	15800
GO TO 50	15900
30 CONTINUE	16000
40 YMAX=ARRAY(J)	16100
PRINT 45,JJ	16200
45 FORMAT (1X,*JJ=*,3)	16300
GO TO 60	16400
50 JJ=J+1	16500
GO TO 10	16600
C SMALLEST NUMBER	16700
60 L=1	16800
LL=1+L	16900
100 DO 300 M=LL,LAST	17000
IF(ARRAY(L).LE.ARRAY(M))300,200	17100
200 L=M	17200
IF(M.EQ.LAST)GO TO 400	17300
GO TO 500	17400
300 CONTINUE	17500
400 YMIN=ARRAY(L)	17600
PRINT 450,LL	17700

450	FORMAT (1X,*LL=*13)	17800
	GO TO 600	17900
500	LL=L+1	18000
	GO TO 100	18100
600	RETURN	18200
	END	18300
	SUBROUTINE SPHJN(X,N,ARR)	18400
C3	NOL BESM Bessel FUNCTIONS OF THE FIRST AND SECOND KINDS	18500
	DIMENSION ARR(1)	18600
	RX=1.0E+307	18700
	NF=1	18800
	NN=N+1	18900
	IF(X.EQ.0.0) GO TO 4	19000
	IF(X.LE.0.5) GO TO 6	19100
C	FOR ORDERS LESS THAN ARGUMENT, AND ARGUMENT GREATER THAN 0.5	19200
	ARR(1)=SIN(X)/X	19300
	ARR(2)=ARR(1)/X-COS(X)/X	19400
	NM=X-2	19500
	IF(N.LT.X) NM=N-1	19600
	IF(NM.LT.0) NM=0	19700
	DO 16 I=1,NM	19800
	FN=I+1	19900
	ARR(I+2)=(FN+1.0)*ARR(I+1)/X-ARR(I)	20000
C	CHECK FOR UNDERFLOW	20100
	IF(ARR(I+2).NE.0.0) GO TO 16	20200
	ARR(I+1)=ARR(I+1)*RX	20300
	ARR(I+2)=(FN+1.0)*ARR(I+1)/X-ARR(I)*RX	20400
	PRINT 13, X, I, N	20500
16	CONTINUE	20600
	IF(N.LT.X) RETURN	20700
	HOLD=ARR(NM+2)	20800
C	FOR ORDERS GREATER THAN OR EQUAL TO ARGUMENT, AND ARGUMENT	20900
C	GREATER THAN 0.5	21000
14	ALPHA=1.0	21100
	M=-2-NM	21200
	NOTE=0	21300
	I=N+47	21400
	IF(X/N.GT.0.9) I=I+X/10.0	21500
	ARR(I+2)=ALPHA	21600
	ARR(I+3)=0.0	21700
	I=-I-1	21800
	DO 2 KK=I,M	21900
	K=-KK	22000
	NOO=K+1	22100
	TKP1=K+K+1	22200
	ARR(K)=TKP1*ARR(K+1)/X-ARR(K+2)	22300
	IF EXPONENT FAULT 18.2	22400
18	TEMPY=ARR(K)=ARR(K+1)/RX*TKP1/X-ARR(K+2)/RX	22500
	ARR(K)=ARR(K)*RX	22600
	IF EXPONENT FAULT 24.2	22700
24	ARR(K)=TEMPY	22800
	ARR(K+1)=ARR(K+1)/RX	22900
	PRINT 13, X, N, NOO	23000
2	CONTINUE	23100
12	C=HOLD/ARR(K)	23200
	IF(C.NE.0.0) GO TO 14	23300
	HOLD=HOLD*RX	23400
	NOTE=NOTE+1	23500
	GO TO 12	23600

14 IF(NOTE.FQ.0)GO TO 17	23700
MULT=307*NOTE	23800
NUL=L-1	23900
PRINT 21,X,N,NUL,MULT	24000
21 FORMAT (1X*FOR ARGUMENT OF *F17.10* AND ORDERS OF *14* TO *14*	24100
1 *SPH.IN EXPONENT WAS INCREASED BY *16)	24200
17 DO 3 I=K,NN	24300
ARR(I)=ARR(I)*C	24400
3 CONTINUE	24500
RETURN	24600
4 ARR(1)=1.0	24700
DO 5 L=2,NN	24800
ARR(L)=0.0	24900
5 CONTINUE	25000
RETURN	25100
C SERIES FOR ARGUMENT LESS THAN OR EQUAL TO 0.5	25200
6 EPS=5.0F-11	25300
CH=CS=DN=DIV=1.0	25400
DO 10 L=1,NN	25500
ITEXP=NEXPA=MEXPA=LFXPA=0	25600
NO=L-1	25700
IF(L.EQ.1) GO TO 7	25800
CS=CS*X/DN	25900
C CHECK FOR UNDERFLOW	26000
IF(CS.NF.0.0) GO TO 7	26100
C SCALE FOR THIS N AND ALL FOLLOWING N	26200
CS=CH*RX*X/DN	26300
NEXPA=NEXPA+1	26400
7 DN=DN+2.0	26500
CH=CS	26600
XKK=0.0	26700
A=CH2=FORTT=1.0/DIV	26800
TN=2.0*NO	26900
FN=-0.5*X*X	27000
8 XKK=XKK+1.0	27100
FD=XKK*(TN+XKK+XKK+1.0)	27200
A=A*FN/FD	27300
C CHECK FOR UNDERFLOW	27400
IF(A.NF.0.0)GO TO 25	27500
C SCALE FOR THIS N AND ALL FOLLOWING N	27600
A=CH2*RX*FN/FD	27700
DIV=DIV/RX	27800
MEXPA=MEXPA+1	27900
FORTT=FORTT*RX	28000
25 CH2=A	28100
TT=A+FORTT	28200
IF(TT.EQ.0.0.OR.ABS(A/TT).LT.EPS) GO TO 9	28300
FORTT=TT	28400
GO TO 8	28500
9 ARR(L)=CS*TT	28600
C CHECK FOR UNDERFLOW	28700
IF(ARR(L).NF.0.0)GO TO 31	28800
C SCALE FOR THIS N AND ALL FOLLOWING N	28900
ARR(L)=CS*RX*TT	29000
DIV=DIV/RX	29100
LFXPA=LFXPA+1	29200
31 ITEXP=NEXPA+MEXPA+LFXPA	29300
IF(ITEXP.FQ.0)GO TO 10	29400
IF(ITEXP.FQ.1)GO TO 29	29500

THIS PROGRAM IS NOT FULLY TESTED  
 AND IS NOT FULLY DOCUMENTED



PRINT 30, MFXPA, MFXPA, LFXPA	29600
30 FORMAT(3X, * SCALE ERROR *3(2X, I10))	29700
20 PRINT 13, X, NO, N	29800
12 FORMAT(3X, *FOR ARGUMENT OF *F17.10* AND ORDERS OF *I4* TO *	29900
1 I4, * SPHYN EXPONENT WAS INCREASED BY 307*)	30000
10 CONTINUE	30100
END	30200
SUBROUTINE SPHYN(X, N, ARR)	30300
DIMENSION ARR(1)	30400
RX=1.0E+307	30500
IF(X.EQ.0.) 5, 3	30600
5 PRINT 4	30700
4 FORMAT(3X, *DO NOT USE ZERO ARGUMENT FOR SPHYN*)	30800
RETURN	30900
FIND Y0 AND Y1	31000
2 ARR(1)=-COS(X)/X	31100
ARR(2)=ARR(1)/X-SIN(X)/X	31200
NN1=N-1	31300
DO 2 K=1, NN1	31400
TKP1=K+K+1	31500
ARR(K+2)=TKP1*ARR(K+1)/X-ARR(K)	31600
CHECK FOR OVERFLOW	31700
IF EXPONENT FAULT 8, 2	31800
8 ARR(K+1)=ARR(K+1)/RX	31900
ARR(K+2)=TKP1*ARR(K+1)/X-ARR(K)/RX	32000
PRINT 7, X, K, N	32100
7 FORMAT(3X, *FOR ARGUMENT OF *F17.10* AND ORDERS OF *I4* TO *	32200
1 I4, * SPHYN EXPONENT WAS DECREASED BY 307*)	32300
2 CONTINUE	32400
END	32500
SUBROUTINE LAYOUT (XRFFR, YRFFR, NN, XRCO, YPCO, NNX, NNY, NUMBER)	32600
COMMON/LABEL/IL, ILX(5), ILY(5), ILH(5), ILR(5), ILC(5), ILARLE(5,5)	32700
COMMON /RANGE/ XMAX, XMIN, YMAX, YMIN, IFRST(5), LST(5), XSIZE	32800
REAL ILX, ILY, ILH, ILR	32900
DIMENSION XRCO(5), YPCO(5)	33000
DIMENSION XRFFR(NN), YRFFR(NN), PLTARRAY(254)	33100
DATA (ISTRT=1)	33200
INITIALIZATION	33300
N=NN	33400
NX=NNX	33500
NY=NNY	33600
IF (XSIZE.EQ.0.) XSIZE=10.	33700
YSIZE=10.	33800
NSVF=N	33900
N=IARSF (N)	34000
IF (NUMBER.EQ.0) NUMBER=1	34100
IF (ISTRT.EQ.2) GO TO 2	34200
CALL PLOTS(PLTARRAY, 254, 13)	34300
ISTRT=2	34400
2 CONTINUE	34500
	34600
LABEL X AXIS	34700
	34800
	34900
DX= (XMAX-XMIN)/XSIZE	35000
IF (DX.NE.1) GO TO 3	35100
CALL SCALE (XRFFR, N, XSIZE, XMIN, DX, 1)	35200
GO TO 7	35300
3 CONTINUE	35400

DO 6 I=1,N	35500
PRINT 5555,XRFFR(I),YRFFR(I),I	35600
5555 FORMAT (2F20.10,110)	35700
XRFFR(I)=(XRFFR(I)-XMIN)/DX	35800
7 CONTINUE	35900
	36000
	36100
	36200
CALL SCALE (XRFFR,N,XSIZE,XMIN,DX,1)	36300
ZERO=XSIZE-ZERO	36400
IF (NSVF) 5,10,10	36500
XMIN=XSIZE*DX-ARSF(XMIN)	36600
DX=-DX	36700
10 CONTINUE	36800
ZERO=(0.0-XMIN)/DX	36900
PRINT 4,DX,XMIN,ZERO	37000
4 FORMAT (*,DX=*,3F20.5)	37100
CALL AXIS (0.0,0.0,XPCD,NX,XSIZE,0.0,XMIN,DX)	37200
PRINT 4,DX,XMIN,ZERO	37300
NX=-NX	37400
CALL AXIS (0.0,YSIZE,XPCD,NX,XSIZE,0.0,XMIN,DX)	37500
	37600
PRINT 4,DX,XMIN,ZERO	37700
LABEL Y AXIS	37800
	37900
	38000
	38100
DY=(YMAX-YMIN)/YSIZE	38200
IF (DY.NE.0.0) GO TO 14	38300
CALL SCALE (YRFFR,N,YSIZE,YMIN,DY,1)	38400
GO TO 15	38500
14 CONTINUE	38600
DO 16 I=1,N	38700
16 YRFFR(I)=(YRFFR(I)-YMIN)/DY	38800
15 CONTINUE	38900
	39000
PRINT 11,DY,YMIN	39100
11 FORMAT (*,DY=*,3F20.4)	39200
CALL AXIS (0.0,0.0,YPCD,NY,YSIZE,90.0,YMIN,DY)	39300
NY=-NY	39400
PRINT 11,DY,YMIN	39500
CALL AXIS (XSIZE,0.0,YPCD,NY,YSIZE,90.0,YMIN,DY)	39600
PRINT 11,DY,YMIN	39700
	39800
	39900
PLOT CURVE	40000
NUMBER1=NUMBR	40100
IF (NUMBER.LT.0) NUMBR=-NUMBER	40200
ND=(N/NUMBR)-1	40300
N2=0	40400
IF (NUMBER.NE.1) GO TO 8	40500
IFIRST(1)=1	40600
ILST(1)=N	40700
8 CONTINUE	40800
DO 150 J=1,NUMBER	40900
IF (NUMBR1.LT.0) GO TO 148	41000
N1=IFIRST(J)	41100
N2=ILST(J)	41200
GO TO 149	41300
148 CONTINUE	
N1=N2+1	

N2=N1*ND	41400
140 CONTINUE	41500
IPN=3	41600
CALL PLOT (XPFFR(1),YBFFR(1),3)	41700
DO 100 I=N1,N2	41800
PRINT 5555,XPFFR(I),YBFFR(I),I	41900
XRF= XPFFR(I)	42000
YRF= YBFFR(I)	42100
IF (NSVF.LT.0) XPFFR(I)=XSIZE-XPFFR(I)	42200
IF (XRF.LT.0.0.OR.XRF.GT.XSIZE) GO TO 100	42300
IF (YRF.LT.0.0.OR.YRF.GT.YSIZE) GO TO 100	42400
CALL PLOT(XRF,YRF,IPN)	42500
IPN=2	42600
100 CONTINUE	42700
150 CONTINUE	42800
IF (IL.FQ.0) GO TO 255	42900
ICT=0	43000
252 ICT=ICT+1	43100
IF (ICT.GT.IL) GO TO 255	43200
CALL SYMBOL(ILX(ICT),ILY(ICT),ILH(ICT),ILABE(1,ICT),ILR(ICT)	43300
1,ILC(ICT))	43400
GO TO 252	43500
255 CONTINUE	43600
CALL PLOT (XSIZE+3,0.0,-3)	43700
PRINT 250	43800
250 FORMAT (* PLOT DONE*)	43900
	44000
	44100
	44200
	44300
	44400
RESTORE THE DATA	44500
DX=ARS(DX)	44600
DY=ARS(DY)	44700
	44800
DO 400 I=1,N	44900
XPFFR(I)=(DX*XPFFR(I))+XMIN	45000
YBFFR(I)=(DY*YBFFR(I))+YMIN	45100
400 CONTINUE	45200
	45300
	45400
XMAX=0.0	45500
XMIN=0.0	45600
YMAX=0.0	45700
YMIN=0.0	45800
	45900
RETURN	46000
END	46100
SUBROUTINE SCALE (X,N,S,YMIN,DY,K)	46200
DIMENSION X(2)	46300
YMAX=X(1)	46400
YMIN=YMAX	46500
NP=N*K	46600
DO 10 I=1,NP,K	46700
IF(YMAX-X(I))5,6,6	46800
5 YMAX=X(I)	46900
6 IF(X(I)-YMIN) 7,10,10	47000
7 YMIN=X(I)	47100
10 CONTINUE	47200
DY=(YMAX-YMIN)/(S)	

DO 20 I=1,N	47300
20 X(I)=(X(I)-YMIN)/DY	47400
PRINT 25,YMIN,YMAX,DY,X(1),X(N)	47500
25 FORMAT(5F25.5)	47600
RETURN	47700
END	47800
SUBROUTINE AXIS (X,Y,BCD,NC,SIZE,THETA,YMIN,DY)	47900
X14=.07	48000
SIGN=1.0	48100
IF (NC) 1,2,2	48200
1 SIGN=-1.0	48300
2 NAC=XABSF(NC)	48400
TH=THETA*0.017453294	48500
N=SIZE+.50	48600
CTH=COSF(TH)	48700
STH=SIN(TH)	48800
TN=N	48900
XP=X	49000
YP=Y	49100
XA=X-.1*SIGN*STH	49200
YA=Y+.1*SIGN*CTH	49300
C IF (THETA-90.0)10,12,12	49400
C 12 XA=-XA	49500
10 CALL PLOT (XA,YA,3)	49600
C	49700
C DO LOOP DRAWS AXIS AND MAKES TIC MARKS	49800
C	49900
DO 20 I=1,N	50000
CALL PLOT (XB,YB,2)	50100
XC=XB+CTH	50200
YC=YB+STH	50300
CALL PLOT (XC,YC,2)	50400
XA=XA+CTH	50500
YA=YA+STH	50600
CALL PLOT (XA,YA,2)	50700
XB=XC	50800
20 YB=YC	50900
CHAR=ABSF (YMIN)	51000
ARSV= ARSF (YMIN+DY)	51100
IF (ARSV-CHAR) 5,6,6	51200
5 ARSV=CHAR	51300
6 FXP=0.0	51400
80 NT=ARSV+0.1	51500
CHAR=NT	51600
IF (CHAR-ARSV) 90,92,92	51700
90 FXP=EXP-1.0	51800
ARSV=ARSV*10.	51900
GO TO 89	52000
91 FXP=FXP+1.0	52100
7 ARSV=ARSV/10.0	52200
NT=ARSV	52300
CHAR=NT	52400
C IF (CHAR-ARSV)93,91,91	52500
IF (ARSV.GT.1) GO TO 91	52600
93 ADY=DY*10.0**(-FXP)	52700
ARSV=YMIN*10.0**(-FXP)+TN*ADY	52800
XA=XB-1.20*SIGN-0.05)*STH-.25*CTH	52900
YA=YB+1.20*SIGN-.05)*CTH-.25*STH	53000
N=N+1	53100

C	IF (THETA-90.) 31,32,32	53200
C	32 XA=-XA+.1	53300
C		53400
C	DO LOOP PLOTS NUMBERS ALONG AXIS	53500
C		53600
	31 DO 30 I=1,N	53700
	CALL NUMRER (XA,YA,.07*ABSV,THETA,4HF7.3)	53800
	ABSV=ABSV-ADY	53900
C	IF (ABSV.LT.0.AND.ABSV.GT. 1) 55,56	54000
C	55 ABSV=0	54100
	56 XA=XA-CTH	54200
	30 YA=YA-STH	54300
	TNC=NAC+7	54400
C		54500
C	CALCULATES CENTER POSITION FOR LABEL	54600
	XA=X+(SIZE/2.0-.06*TNC)*CTH-(-.07+SIGN#.36)*STH	54700
	YA=Y+(SIZE/2.0-.06*TNC)*STH+(-.07+SIGN#.36)*CTH	54800
C	IF (THETA-90.) 41,42,42	54900
C	42 XA=-XA+.1	55000
	41 CALL SYMBOL (XA,YA,X14,PCD,THETA,NAC)	55100
C		55200
C	PRINTS LABEL	55300
	XA=XA+((TNC-6.0)*0.12)*CTH	55400
	YA=YA+((TNC-6.0)*0.12)*STH	55500
	IF (FXP) 35,50,35	55600
	35 CALL SYMBOL (XA,YA,X14,PH(X10),THETA,R)	55700
	XA=XA+.48*CTH-.07*STH	55800
	YA=YA+.48*STH+.04*CTH	55900
	IF (FXP) 40,50,40	56000
C		56100
C	PRINTS SCALE FACTOR	56200
	40 CALL NUMRER (XA,YA,X14,FXP,THETA,4HF3.0)	56300
	50 CONTINUE	56400
	CALL PLOT (XA,YA,3)	56500
	RETURN	56600
	END	56700
	1,GG	
	1,GG	
	1,GG	

### 3-D Version

```

PROGRAM PHASHIFT
C THREE DIMENSIONAL VERSION
C THIS PROGRAM CALCULATES PHASE SHIFTS IN RADIANS OR DEGREES VS E*A**2
C OR K OR E IN DIMENSIONLESS UNITS FOR THE QUANTUM MECHANICAL SCATTERING
C OF A PARTICLE OF ENERGY E FROM A 3-D SQUARE WELL OF WIDTH A AND DEPTH
C -V.
C THE 1ST DATA CARD CONTAINS THE FOLLOWING PARAMETERS...
C THE INPUT PARAMETERS ARE ( 1). EZERO (THE LOWEST ENERGY VALUE) CO
COLS 1-10. (2). EPSILON (THE ENERGY INCREMENT) COLS 11-20. (3). SKIP TO 4
C (4). FERMI (THE WELL WIDTH) COLS 21-30. (5). MAX (THE
C HIGHEST ORDER PHASE SHIFT DESIRED. FOR S.P.D.F) MAX=1,2,3,4. COL 45
C (6). SKIP TO 7 (7). LAST (THE TOT
C AL NUMBER OF POINTS DESIRED TO BE CALCULATED). MAX. 1000. COLS 55-58
C (8). JELLO (IF PRINTOUT OF PHASE SHIFTS IN DEGREES IS DESIRED, ENTER
C 1 IN COL 50. IF RADIANS DESIRED, LEAVE COL 50 BLANK) COL 50
C ENTER THE NUMBER OF CURVES TO BE PLOTTED IN COL 60 (MAX 9).
C IF NO PLOT IS DESIRED, ENTER A 1 IN COL. 65. IF ORDINAL SELF-SCALING
C DESIRED, ENTER A 1 IN COL 70. AND INSERT VALUES OF YMIN AND YMAX BEFO
C T CALL.
C THE 2ND DATA CARD CONTAINS THE WELL DEPTH VULCAN. COLS 1-10. AND KILL
C COL 21. DESIGNATES THE LAST DATA CARD.
COMMON/LABEL/IL,ILX(9),ILY(9),ILH(9),ILR(9),ILC(9),ILARLE(9,9)
COMMON /RANGE/ XMAX,XMIN,YMAX,YMIN,IFRST(9),ILST(9),XSIZE
REAL ILX,ILY,ILH,ILR
REAL N
REAL NZ
COMMON/1/ DELTA(1000),CASTOR(1000),POLLUX(1000),HERCULES(1000),AJA
1X(1000),OLYMPUS(1000),HADES(1000),R(1000),DR(1000),H(1000),DN(1000
2),BZ(1000),DBZ(1000),NZ(1000),GAMMA(1000),ENERGY(1000),CYCLOPS(100
30),CENTAUR(1000),ZFUS(1000),RHO(1000),DNZ(1000)
DIMENSION DELTA1(1000),DELTA2(1000),DELTA3(1000),DELTA
14(1000),W(5)
DIMENSION ERGO(1000)
COMMON/3/ PHASE(1000),ERGO(1000)
KC=-1
700 READ (60,800)EZERO,EPSILON,FERMI,MAX,JELLO,LAST,NCURVES,NO PLOT,NOS
1CALE
800 FORMAT(3(F10.7),14X,I1,4X,I1,4X,I4,1X,I1,4X,I1,4X,I1)
o READ(60,10)VULCAN,KILL
10 FORMAT(F10.5,10X,I1)
IFRST(1)=1
IFRST(2)=1+ LAST
IFRST(3)=1+ 2*LAST
IFRST(4)=1+ 3*LAST
IFRST(5)=1+ 4*LAST
IFRST(6)=1+5*LAST
IFRST(7)=1+6*LAST
IFRST(8)=1+7*LAST
IFRST(9)=1+8*LAST
ILST(1)= LAST
ILST(2)=2*LAST
ILST(3)=3*LAST
ILST(4)=4*LAST
ILST(5)=5*LAST
ILST(6)=6*LAST
ILST(7)=7*LAST
ILST(8)=8*LAST
ILST(9)=9*LAST
XMIN=0.

```

XMAX1=EPSILON*LAST	6000
XTICK=XMAX1/10.	6100
XMAX=(NCURVES-1)*XTICK/2.+XMAX1	6200
XSIZE=XMAX/XTICK	6300
KC=KC+1	6400
ORSHIFTX=XTICK/2.	6500
YMIN=0.	6600
YMAX=1.	6700
YTICK=YMAX1/10.	6800
ORSHIFTY=YTICK/4.	6900
YMAX=(NCURVES-1)*ORSHIFTY+YMAX1	7000
IF (JELLO.EQ.1) GO TO 2	7100
1 PRINT 6	7200
GO TO 3	7300
2 PRINT 7	7400
3 PRINT 12,VULCAN,FERMI,EZERO,XMAX1	7500
PRINT 8	7600
6 FORMAT (1X,**PHASE SHIFTS IN RADIANS FOR A QUANTUM MECHANICAL PARTI	7700
1CLE SCATTERING OFF A 3-D SQUARE WELL.*/)	7800
7 FORMAT (1X,**PHASE SHIFTS IN DEGREES FOR A QUANTUM MECHANICAL PARTI	7900
1CLE SCATTERING OFF A 3-D SQUARE WELL.*/)	8000
8 FORMAT(5X,**ENERGY(1)*4X*DFLTA(0)*4X*DELTA(1)*4X*DELTA(2)*4X*DELTA(	8100
13)*4X*DELTA(4)*4X* K *4X* ERGO *4X*OLYMPUS1*4X*HADES1*/)	8200
12 FORMAT(1X,**WELL DEPTH= *F10.5,** WELL WIDTH= *F10.7,** EZERO= *	8300
1 F10.7,** ELAST= *F10.7/)	8400
K=MAX+1	8500
ENERGY(1)=EZERO	8600
DO 91 I=1, LAST	8700
ERGO(I)=ENERGY(I)*(FERMI)**2	8800
14 ENERGY(I+1)=EPSILON+ENERGY(I)	8900
14 CYCLOPS(I)=SORT(ENERGY(I))	9000
20 CENTAUR(I)=SORT(ENERGY(I)+VULCAN)	9100
24 ZEUS(I)=CYCLOPS(I)*FERMI	9200
ZEUSS=ZEUS(I)	9300
27 IF(ZEUSS.FO.0) 78,30	9400
30 RHO(I)=CENTAUR(I)*FERMI	9500
RHO0=RHO(I)	9600
34 CALL SPHJN(RHO0,K,P)	9700
CALL SPHYN(RHO0,K,N)	9800
DR(1)=-R(2)	9900
DN(1)=-N(2)	10000
DO 35 JI=2,MAX	10100
DB(JI)=(JI*R(JI-1)-(JI+1)*R(JI+1))/(2*JI+1)	10200
DN(JI)=(JI*N(JI-1)-(JI+1)*N(JI+1))/(2*JI+1)	10300
35 CONTINUE	10400
CALL SPHJN(ZEUSS,K,RZ)	10500
CALL SPHYN(ZEUSS,K,NZ)	10600
DRZ(1)=-RZ(2)	10700
DNZ(1)=-NZ(2)	10800
37 DO 38 KI=2,MAX	10900
DRZ(KI)=(KI*RZ(KI-1)-(KI+1)*RZ(KI+1))/(2*KI+1)	11000
DNZ(KI)=(KI*NZ(KI-1)-(KI+1)*NZ(KI+1))/(2*KI+1)	11100
38 CONTINUE	11200
40 DO 76 LI=1,MAX	11300
44 GAMMA(LI)=CENTAUR(I)*DR(LI)/R(LI)	11400
44 CASTOR(LI)=CYCLOPS(I)*DRZ(LI)	11500
50 POLLUX(LI)=GAMMA(LI)*RZ(LI)	11600
54 MERCULES(LI)=CYCLOPS(I)*DNZ(LI)	11700
60 AJAX(LI)=GAMMA(LI)*NZ(LI)	11800

65 OLYMPUS(LI)=CASTOR(LI)-POLLUX(LI)	11900
70 HADES(LI)=HERCULES(LI)-AJAX(LI)	12000
75 W(LI) =ATAN2(OLYMPUS(LI),HADES(LI))	12100
IF(W(LI),LT,0) W(LI)=W(LI)+3.14159	12200
IF(JELLO.EQ,0) GO TO 76	12300
W(LI)=W(LI)*57.29578	12400
76 CONTINUE	12500
IF(JELLO.EQ,1)GO TO 80	12600
W(LI)=W(LI)/3.14159	12700
77 GO TO 80	12800
C THIS SETS DELTA = ZERO FOR E=0	12900
78 DO 79 LI=1,MAX	13000
W(LI)=0.	13100
79 CONTINUE	13200
OLYMPUS(1)=0.	13300
HADES(1)=0.	13400
80 PRINT 85,FNFRGY(1),W(1),W(2),W(3),W(4),W(5),ZEUS(1),ERGO(1),OL	13500
YMPUS(1),HADES(1)	13600
85 FORMAT (10(2X,F11.7))	13700
DELTA(1)=W(1)	13800
DELTA1(1)=W(2)	13900
DELTA2(1)=W(3)	14000
DELTA3(1)=W(4)	14100
DELTA4(1)=W(5)	14200
91 CONTINUE	14300
86 PRINT 87	14400
87 FORMAT (////)	14500
C SURROUTINE MINIMAX AUTOMATICALLY FINDS YMIN AND YMAX FOR LAYOUT.	14600
C YMIN=DELTA(MIN), YMAX=DELTA(MAX), XMIN=EZERO,XMAX= EPSILON*LAST.	14700
IF(NOPLOT.EQ,1) GO TO 95	14800
IF (NOSCALE.EQ,1)GO TO 99	14900
CALL MINIMAX(DELTA,YMIN,YMAX,LAST)	15000
C THE PLOT ROUTINE, LAYOUT, IS DUE TO STEVE BRENNER, CODE )NRL	15100
C XSIZE=PHYSICAL LENGTH OF PLOT IN INCHES. XMIN =MIN. VALUE OF ABSCISSA	15200
C XMAX=MAX VALUE OF ABSCISSA. (XMAX-XMIN)/XSIZE DETS VALUE OF TIC	15300
C MARKS. SIMILARLY Y REFERS TO ORDINATE, YSIZE IS FIXED AT 10 INCHES.	15400
C IN THE CALL FOR SURROUTINE LAYOUT, THE PARAMETERS IN THE CALL STATE-	15500
C MENT ARE RESPECTIVELY ABSCISSA VARIABLE, ORDINATE VARIABLE, NUMBER OF	15600
C DATA POINTS(=LAST),NO. OF HOLLERITH CHARACTERS FOLLOWED BY ABSCISSA	15700
C IDENT.,ORDINATE IDENT. E.G., 8HDELTA=0. (8 CHARACTERS MAX), FOLLOWED	15800
C BY THE ACTUAL NUMBER OF CHARACTERS USED IN THE ABSS.,ORD. NAMES.	15900
C FINALLY THE TOTAL NUMBER OF PLOTS DESIRED SUPERIMPOSED ON THE SAME	16000
C GRAPH. IF ONE PLOT PER GRAPH, USE 1. IF N, USE N. HOWEVER, IF N	16100
C PLOTS ARE SUPERIMPOSED, THE THE DIMENSION STATEMENTS FOR THE ORD,ABS	16200
C AND ALSO LAST MUST BE SCALED BY N. IF SELF-SCALING IS DESIRED, SET	16300
C SURROUTINE MINIMAX AUTOMATICALLY FINDS YMIN AND YMAX FOR LAYOUT.	16400
C YMIN=DELTA(MIN), YMAX=DELTA(MAX), XMIN=EZERO,XMAX= EPSILON*LAST.	16500
C THE PLOT ROUTINE WILL SCALE THE PLOTTED VALUES BY POWERS OF TEN.	16600
C IF SELF-SCALING NOT USED, THEN YMIN AND YMAX VALUES MUST BE SUPPLIED.	16700
C IF DESIRED, DIFFERENT ORDER PHASE SHIFTS MAY BE PLOTTED ON THE SAME	16800
C GRAPH. USE NOSCALE TO BYPASS, PUT A 1 IN COL 70. SUPPLY YMIN,YMAX.	16900
99 CONTINUE	17000
NLAST=NCURVES*LAST	17100
J=LAST	17200
ORSHIFTX=ORSHIFTX*KC	17300
ORSHIFTY=ORSHIFTY*KC	17400
JAZZ=KC*J	17500
DO 90 M=1,J	17600
PHASE(M+JAZZ)=DELTA(M)+ORSHIFTY	17700



ERGON(M+JAZZ)=ERGO(M)+ORSHIFTX	17800
90 CONTINUE	17900
PRINT 100	18000
100 FORMAT(///)	18100
95 IF(KILL.EQ.0) GO TO 9	18200
IF(NOPLOT.EQ.1)GO TO 199	18300
CALL LAYOUT(ERGON,PHASE,NLAST,6HE*A**2,7H-DELTA-,6,7,NCURVES)	18400
CALL STOPPLOT	18500
100 END	18600
SUBROUTINE MINIMAX(ARRAY,YMIN,YMAX,LAST)	18700
DIMENSION ARRAY(1000)	18800
C THIS SUBROUTINE FINDS BOTH THE LARGEST AND SMALLEST NUMBERS IN ARRAY.	18900
C LARGEST NUMBER	19000
J=1	19100
JJ=1+J	19200
10 DO 30 K=JJ, LAST	19300
IF(ARRAY(J).GE.ARRAY(K))30,20	19400
20 J=K	19500
IF(K.EQ.LAST) GO TO 40	19600
GO TO 50	19700
30 CONTINUE	19800
40 YMAX=ARRAY(J)	19900
PRINT 45,JJ	20000
45 FORMAT (1X,*JJ=*13)	20100
GO TO 60	20200
50 JJ=J+1	20300
GO TO 10	20400
C SMALLEST NUMBER	20500
60 L=1	20600
LL=1+L	20700
100 DO 300 M=LL, LAST	20800
IF(ARRAY(L).LE.ARRAY(M))300,200	20900
200 L=M	21000
IF(M.EQ.LAST)GO TO 400	21100
GO TO 500	21200
300 CONTINUE	21300
400 YMIN=ARRAY(L)	21400
PRINT 450,LL	21500
450 FORMAT (1X,*LL=*13)	21600
GO TO 600	21700
500 LL=L+1	21800
GO TO 100	21900
600 RETURN	22000
END	22100
SUBROUTINE SPHJN(X,N,ARR)	22200
C 3 NPL RESN BESSEL FUNCTIONS OF THE FIRST AND SECOND KINDS	22300
DIMENSION ARR(1)	22400
RX=1.0E+307	22500
NF=1	22600
NN=N+1	22700
IF(X.EQ.0.0) GO TO 4	22800
IF(X.LE.0.5)GO TO 6	22900
C FOR ORDERS LFSS THAN ARGUMENT, AND ARGUMENT GREATER THAN 0.5	23000
ARR(1)=SIN(X)/X	23100
ARR(2)=ARR(1)/X-COS(X)/X	23200
NM=X-2	23300
IF(N.LT.X)NM=N-1	23400
IF(NM.LT.0)NM=0	23500
DO 16 I=1,NM	23600

	FN=I+I	23700
	ARR(I+2)=(FN+1.0)*ARR(I+1)/X-ARR(I)	23800
C	CHECK FOR UNDERFLOW	23900
	IF(ARR(I+2).NE.0.0)GO TO 16	24000
	ARR(I+1)=ARR(I+1)*RX	24100
	ARR(I+2)=(FN+1.0)*ARR(I+1)/X-ARR(I)*RX	24200
	PRINT 13, X, I, N	24300
16	CONTINUE	24400
	IF(N.LT.X)RETURN	24500
	HOLD=ARR(NM+2)	24600
C	FOR ORDERS GREATER THAN OR EQUAL TO ARGUMENT, AND ARGUMENT	24700
C	GREATER THAN 0.5	24800
15	ALPHA=1.0	24900
	M=-2-NM	25000
	NOTE=0	25100
	I=N+47	25200
	IF(X/N.GT.0.9)I=I+X/10.0	25300
	ARR(I+2)=ALPHA	25400
	ARR(I+3)=0.0	25500
	I=-I-1	25600
	DO 2 KK=I,M	25700
	K=-KK	25800
	NOO=K+1	25900
	TKP1=K+K+1	26000
	ARR(K)=TKP1*ARR(K+1)/X-ARR(K+2)	26100
	IF EXPONENT FAULT 18.2	26200
18	TEMPY=ARR(K)=ARR(K+1)/RX*TKP1/X-ARR(K+2)/RX	26300
	ARR(K)=ARR(K)*RX	26400
	IF EXPONENT FAULT 24.2	26500
24	ARR(K)=TEMPY	26600
	ARR(K+1)=ARR(K+1)/RX	26700
	PRINT 13, X, N, NOO	26800
	CONTINUE	26900
17	C=HOLD/ARR(K)	27000
	IF(C.NE.0.0)GO TO 14	27100
	HOLD=HOLD*RX	27200
	NOTE=NOTE+1	27300
	GO TO 12	27400
14	IF(NOTE.EQ.0)GO TO 17	27500
	MULT=307*NOTE	27600
	NUL=L-1	27700
	PRINT 21, X, N, NUL, MULT	27800
21	FORMAT (3X*FOR ARGUMENT OF *E17.10* AND ORDERS OF *14* TO *16*	27900
1	*SPHJN EXPONENT WAS INCREASED BY *16)	28000
17	DO 3 I=K,NN	28100
	ARR(I)=ARR(I)*C	28200
3	CONTINUE	28300
	RETURN	28400
4	ARR(1)=1.0	28500
	DO 5 L=2,NN	28600
	ARR(L)=0.0	28700
5	CONTINUE	28800
	RETURN	28900
C	SERIES FOR ARGUMENT LESS THAN OR EQUAL TO 0.5	29000
6	FPS=5.0F-11	29100
	CH=CS=DN=DIV=1.0	29200
	DO 10 L=1,NN	29300
	ITEXP=NEXPA=MEXPA=LFXPA=0	29400
	NO=L-1	29500

COPY AVAILABLE TO DDC DOES NOT  
PERMIT FULLY LEGIBLE PRODUCTION

	IF(L.EQ.1) GO TO 7	29600
	CS=CS*X/DN	29700
C	CHECK FOR UNDERFLOW	29800
	IF(CS.NE.0.0) GO TO 7	29900
C	SCALE FOR THIS N AND ALL FOLLOWING N	30000
	CS=CH*RX*X/DN	30100
	NEXPA=NEXPA+1	30200
7	DN=DN+2.0	30300
	CH=CS	30400
	XKK=0.0	30500
	A=CH2=FORIT=1.0/DIV	30600
	TN=2.0*NO	30700
	FN=-0.5*X*X	30800
8	XKK=XKK+1.0	30900
	FD=XKK*(TN+XKK+XKK+1.0)	31000
	A=A*FN/FD	31100
C	CHECK FOR UNDERFLOW	31200
	IF(A.NE.0.0) GO TO 25	31300
C	SCALE FOR THIS N AND ALL FOLLOWING N	31400
	A=CH2*RX*FN/FD	31500
	DIV=DIV/RX	31600
	MFXPA=MEXPA+1	31700
	FORTT=FORTT*RX	31800
25	CH2=A	31900
	TT=A+FORTT	32000
	IF(TT.EQ.0.0.OR.ARS(A/TT).LT.EPS) GO TO 9	32100
	FORTT=TT	32200
	GO TO 8	32300
9	ARR(L)=CS*TT	32400
C	CHECK FOR UNDERFLOW	32500
	IF(ARR(L).NE.0.0) GO TO 31	32600
C	SCALE FOR THIS N AND ALL FOLLOWING N	32700
	ARR(L)=CS*RX*TT	32800
	DIV=DIV/RX	32900
	LEXPA=LEXPA+1	33000
31	ITEXP=NFXPA+MEXPA+LEXPA	33100
	IF(ITEXP.EQ.0) GO TO 10	33200
	IF(ITEXP.EQ.1) GO TO 29	33300
	PRINT 30, NEXPA, MEXPA, LEXPA	33400
30	FORMAT(3X, 'SCALE ERROR *3(3X, L10)')	33500
29	PRINT 13, X, NO, N	33600
12	FORMAT(3X, 'FOR ARGUMENT OF *E17.10* AND ORDERS OF *I4* TO *	33700
1	I4* SPHJH EXPONENT WAS INCREASED BY 307*)	33800
10	CONTINUE	33900
	END	34000
	SUBROUTINE SPHYN(X, N, ARR)	34100
	DIMENSION ARR(1)	34200
	RX=1.0E+307	34300
	IF(X.EQ.0.) 5, 3	34400
5	PRINT 4	34500
4	FORMAT(3X, 'DO NOT USE ZERO ARGUMENT FOR SPHYN*')	34600
	RETURN	34700
C	FIND Y0 AND Y1	34800
3	ARR(1)=-COS(X)/X	34900
	ARR(2)=ARR(1)/X-SIN(X)/X	35000
	NM1=N-1	35100
	DO 2 K=1, NM1	35200
	TKP1=K+K+1	35300
	ARR(K+2)=TKP1*ARR(K+1)/X-ARR(K)	35400

C	CHECK FOR OVERFLOW	35500
	IF EXPONENT FAULT 8.2	35600
	ARR(K+1)=ARR(K+1)/RX	35700
	ARR(K+2)=TKP1*ARR(K+1)/X-ARR(K)/RX	35800
	PRINT 7,X,K,N	35900
	7 FORMAT(3X,'FOR ARGUMENT OF *F17.10*' AND ORDERS OF *I4*' TO *,	36000
	1 I4,' SPHYN EXPONENT WAS DECREASED BY 307*')	36100
	2 CONTINUE	36200
	END	36300
	SUBROUTINE LAYOUT (XBFFR,YBFFR,NN,XRCD,YBCD,NNX,NNY,NUMBER)	36400
	COMMON/LABEL/IL,ILX(9),ILY(9),ILH(9),ILR(9),ILC(9),ILABLE(9,9)	36500
	COMMON /RANGE/ XMAX,XMIN,YMAX,YMIN,IFIRST(9),ILST(9),XSIZE	36600
	REAL ILX,ILY,ILH,ILR	36700
	DIMENSION XRCD(5),YBCD(5)	36800
	DIMENSION XBFFR(NN),YBFFR(NN),PLTARRAY(254)	36900
	DATA (1STRT=1)	37000
C	INITIALIZATION	37100
	N=NN	37200
	NX=NNX	37300
	NY=NNY	37400
	IF (XSIZE.FQ.0.0) XSIZE=10.	37500
	YSIZE=10.	37600
	NSVE=N	37700
	N=XARSF (N)	37800
	IF (NUMBER.FQ.0) NUMBER=1	37900
	IF (1STRT.FQ.2) GO TO 2	38000
	CALL PLOTS(PLTARRAY,254,13)	38100
	1STRT=2	38200
	2 CONTINUE	38300
C		38400
C	LABEL X AXIS	38500
C		38600
		38700
	DX= (XMAX-XMIN)/XSIZE	38800
	IF (DX.NE.0) GO TO 3	38900
	CALL SCALE (XBFFR,N,XSIZE,XMIN,DX,1)	39000
	GO TO 7	39100
	3 CONTINUE	39200
	DO 6 I=1,N	39300
	PRINT 5555,XBFFR(I),YBFFR(I),I	39400
	5555 FORMAT (2F20.10,I10)	39500
	6 XBFFR(I)= (XBFFR(I)-XMIN)/DX	39600
	7 CONTINUE	39700
		39800
		39900
C	CALL SCALE (XBFFR,N,XSIZE,XMIN,DX,1)	40000
	ZERO=XSIZE-ZERO	40100
	IF (NSVE) 5,10,10	40200
	5 XMIN=XSIZE*DX-ARSF(XMIN)	40300
	DX=-DX	40400
	10 CONTINUE	40500
	ZERO=(0.0-XMIN)/DX	40600
	PRINT 4,DX,XMIN,ZERO	40700
	4 FORMAT (4,DX,XMIN,ZERO)	40800
	CALL AXIS (0.0,0.0,XBCD,NX,XSIZE,0.0,XMIN,DX)	40900
	PRINT 4,DX,XMIN,ZERO	41000
	NX=-NX	41100
	CALL AXIS (0.0,YSIZE,XBCD,NX,XSIZE,0.0,XMIN,DX)	41200
C		41300

PRINT 4,DX,XMIN,ZERO	41400
C LABEL Y AXIS	41500
C	41600
	41700
	41800
DY= (YMAX-YMIN)/YSIZE	41900
IF (DY.NE.0.0) GO TO 14	42000
CALL SCALE (YBFFR,N,YSIZE,YMIN,DY,1)	42100
GO TO 15	42200
14 CONTINUE	42300
DO 16 I=1,N	42400
16 YBFFR(I)=(YBFFR(I)-YMIN)/DY	42500
15 CONTINUE	42600
	42700
	42800
PRINT 11,DY,YMIN	42900
11 FORMAT (1X,DY=,3F20.4)	43000
CALL AXIS (0.0,0.0,YBCD,NY,YSIZE,90.,YMIN,DY)	43100
NY=-NY	43200
PRINT 11,DY,YMIN	43300
CALL AXIS (XSIZE,0.0,YBCD,NY,YSIZE,90.,YMIN,DY)	43400
PRINT 11,DY,YMIN	43500
C	43600
C PLOT CURVE	43700
NUMBER1=NUMBER	43800
IF (NUMBER.LT.0) NUMBER=-NUMBER	43900
ND=(N/NUMBER)-1	44000
N2=0	44100
IF (NUMBER.NE.1) GO TO 8	44200
IFIRST(1)=1	44300
ILST(1)=N	44400
8 CONTINUE	44500
DO 150 J=1,NUMBER	44600
IF (NUMBER1.LT.0) GO TO 148	44700
N1=IFIRST(J)	44800
N2=ILST(J)	44900
GO TO 149	45000
148 CONTINUE	45100
N1=N2+1	45200
N2=N1+ND	45300
149 CONTINUE	45400
IPN=3	45500
C	45600
C CALL PLOT (XBFFR(1),YBFFR(1),3)	45700
C	45800
DO 100 I=N1,N2	45900
C PRINT 555,XBFFR(I),YBFFR(I),I	46000
XB= XBFFR(I)	46100
YB= YBFFR(I)	46200
IF (NSVF.LT.0) XBFFR(I)=XSIZE-XBFFR(I)	46300
IF (XRF.LT.0.0.OR.XRF.GT.XSIZE) GO TO 100	46400
IF (YRF.LT.0.0.OR.YRF.GT.YSIZE) GO TO 100	46500
CALL PLOT(XRF,YRF,IPN)	46600
IPN=2	46700
100 CONTINUE	46800
150 CONTINUE	46900
IF (IL.FQ.0) GO TO 255	47000
ICT=0	47100
255 ICT=ICT+1	47200

IF (ICT.GT.IL) GO TO 255	47300
CALL SYMBOL(ILX(ICT),ILY(ICT),ILH(ICT),ILABLE(1,ICT),ILR(ICT)	47400
1,ILC(ICT))	47500
GO TO 252	47600
255 CONTINUE	47700
CALL PLOT (XSIZE+3,0,0,-3)	47800
PRINT 250	47900
250 FORMAT (* PLOT DONE*)	48000
C	48100
C	48200
C	48300
RESTORE THE DATA	48400
DX=ARS(DX)	48500
DY=ABS(DY)	48600
C	48700
DO 400 I=1,N	48800
XBFFR(I)=(DX*XBFFR(I))+XMIN	48900
YBFFR(I)=(DY*YBFFR(I))+YMIN	49000
400 CONTINUE	49100
C	49200
XMAX=0.0	49300
XMIN=0.0	49400
YMAX=0.0	49500
YMIN=0.0	49600
C	49700
RETURN	49800
END	49900
SUBROUTINE SCALE (X,N,S,YMIN,DY,K)	50000
DIMENSION X(7)	50100
YMAX=X(1)	50200
YMIN=YMAX	50300
NP=N*K	50400
DO 10 I=1,NP,K	50500
IF (YMAX-X(I))5,6,6	50600
5 YMAX=X(I)	50700
6 IF (X(I)-YMIN) 7,10,10	50800
7 YMIN=X(I)	50900
10 CONTINUE	51000
DY=(YMAX-YMIN)/(5)	51100
DO 20 I=1,N	51200
20 X(I)=(X(I)-YMIN)/DY	51300
PRINT 25,YMIN,YMAX,DY,X(1),X(N)	51400
25 FORMAT(5F25.5)	51500
RETURN	51600
END	51700
SUBROUTINE AXIS (X,Y,BCD,NC,SIZE,THETA,YMIN,DY)	51800
X14=.07	51900
SIGN=1.0	52000
IF (NC) 1,2,2	52100
1 SIGN=-1.0	52200
2 NAC=XABSF(NC)	52300
TH=THETA*0.017453294	52400
N=SIZE+.50	52500
CTH=COSF(TH)	52600
STH=SINF(TH)	52700
TN=N	52800
XB=X	52900
YB=Y	53000
XA=X-.1*SIGN*STH	53100

YA=Y+.1*SIGN*CTH	53200
C IF (THETA-90.) 10.12.12	53300
C 12 XA=-XA	53400
10 CALL PLOT (XA,YA.3)	53500
C	53600
C DO LOOP DRAWS AXIS AND MAKES TIC MARKS	53700
C	53800
DO 20 I=1,N	53900
CALL PLOT (XR,YR.2)	54000
XC=XR+CTH	54100
YC=YR+STH	54200
CALL PLOT (XC,YC.2)	54300
XA=XA+CTH	54400
YA=YA+STH	54500
CALL PLOT (XA,YA.2)	54600
XR=XC	54700
20 YR=YC	54800
CHAR=ABSF (YMIN)	54900
ARSV= ARSF (YMIN+DY)	55000
IF (ARSV-CHAR) 5.6.6	55100
5 ARSV=CHAR	55200
6 EXP=0.0	55300
80 NT=ARSV+.1	55400
CHAR=NT	55500
IF (CHAR-ARSV) 90.92.92	55600
90 EXP=EXP+1.0	55700
ARSV=ARSV*10.	55800
GO TO 89	55900
91 EXP=EXP+1.0	56000
92 ARSV=ARSV/10.0	56100
NT=ARSV	56200
CHAR=NT	56300
IF (CHAR-ARSV) 93.91.91	56400
IF (ARSV.GT.1) GO TO 91	56500
93 ADY=DY*10.0**(-EXP)	56600
ARSV=YMIN*10.0**(-EXP)+TN*ADY	56700
XA=XR-(.20*SIGN-.05)*STH-.25*CTH	56800
YA=YR+(.20*SIGN-.05)*CTH-.25*STH	56900
N=N+1	57000
C IF (THETA-90.) 31.32.32	57100
C 32 XA=-XA+.1	57200
C	57300
C DO LOOP PLOTS NUMBERS ALONG AXIS	57400
C	57500
31 DO 30 I=1,N	57600
CALL NUMBER (XA,YA,.07*ARSV,THETA.4HF7.3)	57700
ARSV=ARSV-ADY	57800
C IF (ARSV.LT.0.AND.ARSV.GT.-1) 55.56	57900
C 55 ARSV=0	58000
56 XA=XA-CTH	58100
30 YA=YA-STH	58200
TNC=NAC+7	58300
C	58400
C CALCULATES CENTER POSITION FOR LABEL	58500
XA=X+(SIZE/2.0-.06*TNC)*CTH-(-.07*SIGN*.36)*STH	58600
YA=Y+(SIZE/2.0-.06*TNC)*STH+(-.07*SIGN*.36)*CTH	58700
C IF (THETA-90.) 41.42.42	58800
C 42 XA=-XA+.1	58900
41 CALL SYMRCL (XA,YA,X14,RCD,THETA,NAC)	59000

C		59100
C	PRINTS LABEL	59200
	XA=XA+((TNC-6.0)*0.12)*CTH	59300
	YA=YA+((TNC-6.0)*0.12)*STH	59400
	IF (EXP) 35,50,35	59500
	35 CALL SYMBOL (XA,YA,X14,RH(X10),THETA,8)	59600
	XA=XA+.48*CTH-.07*STH	59700
	YA=YA+.48*STH+.04*CTH	59800
	IF (EXP) 40,50,40	59900
C		60000
C	PRINTS SCALE FACTOR	60100
	40 CALL NUMRER (XA,YA,X14,EXP,THETA,4HF3.0)	60200
	50 CONTINUE	60300
	CALL PLOT (XA,YA,3)	60400
	RETURN	60500
	END	60600
	!GG	
	!GG	
	!GG	